ACADEMIC PORTFOLIO

Computer Science & Engineering

Missouri University of Science & Technology

Missouri S&T

Comprehensive Academic Journey

A curated collection of coursework, projects, and research spanning foundations through advanced graduate studies

Course Coverage

Computer Science:	8 Courses (CS 1200-5400)
Mathematics:	4 Courses (MATH 1214-2222)
Engineering Sciences:	6 Courses (CPE/PHYS)
Liberal Arts:	3 Courses (Ethics, Psychology, Statistics)
Teaching Materials:	Instructional Documents

Compiled: June 30, 2025

Generated with XeLaTeX & Claude Code

Missouri S&T - Miners Excel

Contents

Introduction

This document represents a comprehensive academic portfolio from Missouri University of Science & Technology, containing coursework across multiple computer science and related disciplines. The repository was consolidated from separate course repositories using a custom git consolidation script that preserves commit history.

Repository Structure

The academic work is organized by course using the pattern `{department}{number}-{course-na`
Each course contains assignments, projects, lecture notes, and other academic materials in their original formats and styles.

About This Compilation

This master document aggregates individual academic documents while preserving their original formatting and styles. Each included document maintains its own headers, styling, and content structure as originally created.

Part I

Computer Science

Chapter 1

CS 1510 - Data Structures

Graph Algorithms

Course: `../cs1510-data-structures/Graph.pdf`

Document 1

Included from: `../cs1510-data-structures/Graph.pdf`

1 Graph

Is there a path from root a to node b? Let's algorithmize!

```
// Recursive
pathSearch(Graph G, start, goal, visited nodes) {
    if start == goal { return true; }

    add started to visited nodes
    for every neighbor x of start not in visited nodes
        solved = pathSearch(G, x, goal, visited nodes)
        if solved {
            return true
        }

    return false;
}

// Not Recursive
pathSearch(Graph G, srat, goal, Visited) {
    stack of nodes S
    push(S, start)

    while (S is not empty) {
        X = top(S)
        pop(S);

        if (x == goal)
            return true

        add x to visited

        for every neighbor y of x, not in viisted
            push (S, y)
    }

    return false;
}
```

Heap Data Structure

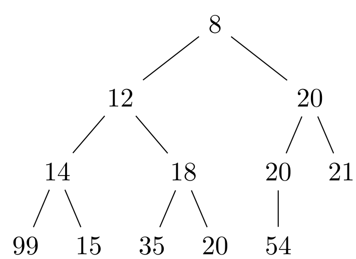
Course: `../cs1510-data-structures/Heap.pdf`

Document 2

Included from: `../cs1510-data-structures/Heap.pdf`

Heap

- Binary Search Tree
- All but the bottom level is complete.
- From every node x, x is less than it's two children.
- Member functions.
  - top(T) $= 8$
  - insert(T, x)
  - remove(T) $= T =$ *the top of the heap*
- Maintenance Functions
  - Percolate Up, during insertion.
    - Let the item bubble up.
  - Percolate Down, during removal.
    - Like a stone sinking through a viscous liquid.



D.S. ArrayHeap

```
class ArrayHeap {
    T *data;
    int m_max, m_size;

public:
    const T& top() {
        if (m_size != 0) {
            return m_data[0];
        } else {
            cerr << "Shit."
        }
    }

    void insert(const T& x) {
```

```
        if (m_max == m_size) {
            grow();
        }

        int hole = m_size;
        m_size++;

        while (hole > 0 && x < m_data[(hole- 1) / 2]) {
            m_data[hole] = m_data[(hole - 1) / 2];
            hole = (hole - 1)/2;
        }

        m_data[hole] = x;
    }

    void remove() {
        if (m_size == 0) { return; }
        int hole = 0;
        m_size--;

        We can continue this over break.
    }
};
```

Tree Data Structure

Course: `../cs1510-data-structures/Tree.pdf`

Document 3

Included from: `../cs1510-data-structures/Tree.pdf`

1 Tree

- A tree is a collection of elements with a hierarchical relation over such elements.
- The actual definition is as follows: The empty collection is a tree (the empty tree). A single element is usually called a node. Node is a tree.
- If n is a node and $T_1, T_2, T_3, \ldots, T_n$ are trees, then n related to $T_1, T_2, \ldots, T_n$ is drawn:
  - n is called the root
  - $T_1 \ldots T_n$ are called the subtrees of n.
- A *path* is a sequence of nodes. $< a_1, a_2, a_3, \ldots, a_n >$ when n_{i+1} is a parent of $n_i 0 \leq i < n$.
- The *depth* of a node a is the number of nodes in the path from a to the root.
- the *height* of a tree is the greatest depth of a node in the tree.

Note Every tree has only one root.

Child The root of each subtree T_1, T_n are called the children of n

n Is called the parent of the root of each subtree T_1, T_n

Siblings If two roots have the same parent they are called siblings.

Leaf A leaf is a node with no children.

Degree The degree of a node a is the number of children of a.

- The degree of a tree is the highest degree of a node in the tree.

decedent/ancestor If there is a path from node a to node z then a is called a decedent of z. z is called an ancestor of a. The root is every node's ancestor.

1.1 Binary Search Tree

- Binary: Degree 2
- Search Conditions
  - find(T,x)
  - getMin(T)
  - getMax(T)
  - insert(T)
  - remove(T,x)

```
// Data Structure BinaryTree

template <classname T>
class TreeNode {
    T m_data;
    TreeNode *m_right;
    TreeNode *m_left;
};

// To use recursion, functions cannot be a method of TreeNode
const T& getMin(TreeNode *t) {
    if (t == nullptr) { /* error */ }

    if (t -> m_left == nullptr) {
        return t-> m_data;
    } else {
        return getMin(t -> m_left);
    }
}

const T& getMax(TreeNode *t) {
    if (t == nullptr) { /* error */ }
    TreeNode *p = t;

    while (p -> m_right != nullptr) {
        p = p -> m_right;
    }

    return p -> m_data;
}

bool T& find(TreeNode *t, const T& x) {
    if (t == nullptr) { return false; }
    if (t -> m_data == x) { return true; }

    if (x < t -> m_data) {
        return find(t - > m_left, x);
    } else if (x > t -> m_right) {
        return find(t -> right, x);
    }
}

void insert(TreeNode * &t, const T& x) {
    if (t == nullptr) {
        t = new TreeNode;
        t -> m_right = nullptr;
```

```
        t -> m_left = nullptr;
    } else (x < t -> m_data) {
        insert(t -> m_left);
    } else if (x > t -> m_data) {
        insert(t -> m_right);
    } else {
        return; // This is a duplicate. No duplicates allowed.
    }
}

void remove(TreeNode * &t, const T& x) {
/*
3 Cases:
- No Children
- One Child
- Two Children

To remove, you have choice. Max of Left or Min of right.
*/

    if (t == nullptr) {
        return;
    }
    if (x < t -> m_data) {
        remove(t -> left, x);
    } else if (x > t -> m_data) {
        remove(t -> right x);
    } else {
        // FOUND X!
        if (t -> m_right == nullptr && t -> m_left == nullptr) {
            // No children
            delete t;
            t = nullptr;
        } else if (t -> m_right == nullptr || t-> m_left == nullptr) {
            TreeNode *temporary = t -> m_right;
            if (temporary == nullptr) {
                temporary = t -> left;
            }

            // Now, temporary points to the chlid
            delete x;
            t = temporary;
        } else {
            // X has two children
            t -> m_data = getMin(t -> m_right);
            remove(t -> m_right, t -> m_data);
```

```
        }
    }
}
```

- collection of objects
- repetition is not allows
  - SETS!
- Why? Who cares? WHY NOT VECTORS?
  - find()
    - Find of size 500. $log_2 500 = 8.9$
    - 5000. $log_2 5000 = 12.2$
    - 5 Million. $log_2 5 Million = 22.25$
    - Most important operations.
  - insert()
    - $log_2 n$
  - remove()
    - $log_2 n$

index at i, right(i) $= 2i + 1$, left(i) $= 2i + 2$. The parent of is $\frac{i-2}{2}$

Recursion

Course: `../cs1510-data-structures/Recurrsion.pdf`

Document 4

Included from: `../cs1510-data-structures/Recurrsion.pdf`

- Homework 5 is on stacks
- Go to megaminer.

1 Recursive Object

An object which potentially consists or defined in terms of itself. Recursion is used to define things such as:

- Sets
- Functions
- Other objects

Power of Recursion:

- Describe an infite object
- Through finite means.

Recursive Definition

- Base Case
- Recursive Case

Example: Set of all strings of balanced parenthesis.
Base Case: () is in the set.
Recursive Case: if s is in the set, then s(), ()s, and (s) are also in the set.

$$Fibonacci = \begin{cases} fib(1) = 1 \\ fib(2) = 1 \\ fib(n) = fib(n-1) + fib(n-2) \end{cases} \tag{1}$$

$$Factorial = \begin{cases} 1! = 1 \\ n! = n \times (n-1) \end{cases} \tag{2}$$

1.1 Recursive Algorithms

- Base Case
  - Direct solution to a small problem instance.
- Recursive Case
  - Decompose problem into smaller instances.
  - Solve smaller instances.
  - Construct solution from smaller solutions.

1.1.1 Triomino Problem

Suppose we have four possible tiles made of three squares.

Problem: Cover $2^n \times 2^n$ board, where one tiles is a hole with triominoes.

- $n = 4, 2^n = 16$

  Morales gives example.

- split board in 4 equal parts.
- Place tronmino across 3 split parts without a hole.
- Solve each subpart.

```
void foo() {
  int x;

  foo();
}

quicksort(array, left, right) { // assuming left < or = right
  if (left = right) {
    return; // Base Case
  }

  pivot = a[(left + right) / 2];
```

```
  int i = left;
  int j = right;

  repeat
    while (a[i] < pivot) { i++; }
    while (a[j] > pivot) { j--; }
    if (i < j) {
      swap(a[i], a[j]);
      i++;
      j--;
    }
  while (j > i);

  quicksort(a, i + 1, r);
  quicksort(a, l, i - 1)

}
```

2 Recursive Backtracking

```
try
    intialize choices
    do
        select choice
        if choice is valid
            record choice
            if solution complete
                success!
            else
                try next step
                if next step succeeds
                    success!
                else
                    cancel record
    while !success & more choices available

path_find(grid, int row, int col) {
```

```
    for choice c in {N, NE, E}
        nrow = row after c;
        ncol = col after c;

        if (grid[nrow][ncolumn]) != obstacle && nrow, ncol is in bounds)
        record nrow, ncol;

        if (grid[row][column] == cake!) {
            return true;
        } else {
            solve = path_find(grid, nrow, ncol)
            if (solve) {
                return true;
            } else {
                record C;
            }
        }

    return false
}

bool valid(grid, int r, int c) {
    if (c < 0 || r >= N) {
        return false;j
    }
    if (c < 0 || c >= N) {
        return false;
    }
    if (grid[r, c] = obstacle) {
        return false;
    }

    return true;
}

for (int i = 0; i < 3; c++) {
    nrow = col + dir[c][0];
    ncol = col + dir[c][1];
```

```
    if (valid(grid, nrow, ncol, n)
}
```

Chapter 2

CS 3800 - Operating Systems

OS Lecture 1

Course:
`../cs3800-operating-systems/lecture-1.pdf`

Document 5

Included from: `../cs3800-operating-systems/lecture-1.pdf`

1 Introduction

- Interrupts can come about by many sources
  - Keyboard, system clock, etc.
- Interrupts do not effect exceptions
- There is a bit in the PSW to determine if user/superuser mode

OS Lecture 2

Course:
`../cs3800-operating-systems/lecture-2.pdf`

Document 6

Included from: `../cs3800-operating-systems/lecture-2.pdf`

2 An Overview

- Horizontal — Generalized
- Vertical — Specialized

OS Exam 2 Cheatsheet

Course:
`../cs3800-operating-systems/exams/exam-2/cheatsheet.pdf`

Document 7

Included from: `../cs3800-operating-systems/exams/exam-2/cheatsheet.pdf`

1 Memory Management

1.1 Requirements

- Relocation: Programmer/compiler does not know where program will be in memory. Program may also be moved around during execution. Memory references in the code must be translated.
- Protection: Processes shouldn't be allowed to access other's memory without permission. Impossible to know values since program can be relocated, which requires runtime checks. OS cannot anticipate all accesses of a program.
- Sharing: Allow several processes to share the same portion of memory. Better to allow each process access to the same copy of a program than have separate copies.

1.2 Methods - Fixed Partitioning

- Memory is divided into partitions which are assigned on demand. These can be equal or variable-sized.
- Inefficient, any program, no matter how small, occupies and entire partition.
- Strategies: Equal-sized requires no strategy. Unequal typically assigns each process to the smallest partition which it can fit, wasting the least amount of memory.
- Partition Assignment: Can give each partition a queue or have a main queue for the entire memory for processes to wait.

1.3 Methods - Dynamic Partitioning

- Processes are allocated exactly as much memory as required.
- Eventually holes start appearing in memory (unallocated bits when processes finish).
- External Fragmentation: Chunks of unallocated memory left from processes finishing. When all over, processes can't find space.
- Internal Fragmentation: With fixed page sizes, the last page of a program that can be partially filled.
- Must use compaction to shift processes so they are contiguous and free memory is in one block.

1.4 Strategies for Dynamic Partitioning

- First-fit: fastest, however can have many processes allocated at front end that must be searched over while searching for a new block. Looks for the first block of memory that fit.
- Best-fit: Worst performer, chooses block that is closest in size to request. Causes more computation to be required as external fragments are the smallest too.
- Next-fit: Like first fit, but searches for free blocks beyond the last (time) allocated block. Results in event distribution of free block in memory.

1.5 Methods - Buddy System

- Entire space is treated as a single block of 2^u. If a request of size s is such that $2^{u-1} < s \leq 2^u$ then the entire block of 2^u is allocated.
- Otherwise, split into two equal buddies.
- Splitting continues until smallest block $\geq$ s is generated.
- Data structure resembles a binary tree.

2 Virtual Memory

- Page Fault: Page table indicates that virtual address isn't in memory, os exception handler is invoked to move data from disk to memory.
- Translation Lookaside Buffer: High speed cache to look up page table entries. Stores most recently used page table entries. Uses associative mapping (many page numbers can map to the same TLB index).
  - Given a virtual address, the processor examines the TLB.
  - If a page table entry is present, it's a "hit" and the frame number is returned which leads to the read address.
  - If it isn't ("miss"), the page number is used to look it up, and the TLB is updated with this data.

2.1 Page Sizes

- Multi-level page tables allow for a new page table above another, which then maps to other entries.
- Smaller page side leads to less internal fragmentation, larger page tables, and page tables ending up in virtual memory (as they are so large). This can, in turn, lead to double page faults.
- Small page sizes means more pages fit in memory, leading to fewer page faults. This is due in part to smaller pages that are used frequently staying in memory, instead of large chunks with a small bit used taking up space.
- Secondary memory is designed to transfer large chunks of data efficiently, which favors large page sizes.
- Larger page sizes leads to some useless references being in memory taking up space.

3 Uniprocessor Scheduling

3.1 Aims

- Response Time: time it takes a system to reactive to a given input (reduce).
- Turnaround time: (TAT) total time spent in the system, waiting time + service time.
- Throughput: jobs per minute (inverse of TAT), maximize.

3.2 Types

- Long-Term Scheduling
  - Determines which programs are admitted to system for processing
  - Controls degree of multiprogramming
  - Which job to admit? FCFS, Priority, expected exec time, I/O reqs
- Medium-Term Scheduling
  - Part of the swapping function
  - Swapping-in decision is based on the need to manage the degree of multiprogramming
- Short-term scheduling is the dispatcher. It executes most frequently and is invoked when events occur. Including clock interrupts, I/O, OS calls, signals, etc.

3.3 Factors

- Priorities: scheduler should choose higher priority processes over lower priority ones. Uses many ready queues to represent each level. Lower priority processes can suffer starvation.
- Decision Modes:
  - Nonpreemptive: Once a process is in the running state, it will continue until termination or blocks self.
  - Preemptive: Currently running process can be interrupted and moved to the ready state due to external event. No single process can monopolize processor for long.

3.4 Schedulers

- First-Come-First-Serve (FCFS): nonpreemptive scheduler where oldest process is scheduled to run next.
  - Advantage: favors CPU-bound processes. I/O processes wait for CPU bound ones to finish.
  - Disadvantage: Short processes can wait for a long time before running.
- Round Robin (RR): preemptive based on clock (time sliced) interrupt at regular intervals. When an interrupt occurs, process is placed into ready and next job runs.
- Shortest Process Next (SPN): Nonpreemptive, process with shortest expected processing time is next. For batch jobs, user is required to estimate the running time . For interactive jobs, OS predicts it. Short processes get priority here and long ones can be starved.
- Shortest Remaining Time (SRT): Preemptive version of $\wedge$. Achieves better turnaround time to $\wedge$ as a short job is given immediate preference to a long one. Hard to estimate remaining time.
- Highest Response Ratio Next (HRRN): Nonpreemptive, uses R= (time spent waiting + service time) / service time decide next process. No starvation possible, shorter jobs preferred.

4 Multiprocessing

4.1 Issues

- Multiprogramming usage— should we allow one application to lock up several cores (maximum speedup)?
- Unless a single queue is used for scheduling, it becomes more difficult to maintain specific disciplines.
- When a single queue is used, a single process can be schedule to run on any processor (master node needed).
- Preemptive schemes (RR) are costly to implement with a single queue approach.

4.2 Real-time systems

- Tasks or processes attempt to control/react to events which must keep up.
- Correctness depends on both result AND time at delivery.
- Critical that the system is reliable, sometimes with failsafe (traffic lights).
- Sometimes include small size, fast context switches, prioritization of scheduling, and special alarms/timeouts.
- Hard real-time task: must meet the deadline, causes catastrophic failure/errors w/o.
- Soft real-time task: has a deadline that is desired. Makes sense to complete even if late.
- Periodic tasks: Only one unit per period T, exactly T apart.
- Aperiodic task: has a deadline by which it must finish/start/both.

4.3 Deadline scheduling

- Real time application, not concerned with fairness/response time but with prioritizing tasks based on deadlines.
- Earliest deadlines typically scheduled first.
- Best to know each first:
  - Ready time (periodic tasks know this)
  - Starting/completion deadline
  - Processing time and resource reqs.
  - Priority

4.4 Rate Monotonic Scheduling

- Assigns priorities to tasks on basis of periods.
- Highest priority tasks have the shortest period.
- Always works if the below is satisfied:
  $\frac{C_1}{T_1} + \frac{C_2}{T_2} + \cdots + \frac{C_n}{T_n} \leq n(2^{\frac{1}{n}} - 1)$
- Has industrial applications.
- Higher priority tasks have higher frequencies (hz)
- n is the number of tasks, C is cycle time T is processing time?

5 Networking

5.1 Internet Protocol

- Uniform method for host addresses, each host getting a unique one.
- Provides packet delivery mechanisms (Packet: standard transfer unit), packets having a header with destination and size and a payload.
- ISO Open Systems Interconnect Model has 7 layers. From top to bottom: Application, presentation, session, transport, network, data link, physical.
- ISO OSI is a framework for specific protocols (FTP/RPC/TCP).

5.2 Low Level Protocols

- Physical layer is the signaling tech— all hardware.
- Data link layer, frame management. MAC addresses, form of hexadecimal letters X XX:XX:XX:XX:XX:XX, 6 pairs.
- Broadcasts are on 6 pairs of FF.
- Examples are wireless, Ethernet, and X.25.
- These addresses are 48 bit.
- Hosts send bits to other hosts in frames.
- Hubs take one frame and send it to every other port.

5.3 OSI Layers (cont)

- Network layer
  - Combines networks using IP.
  - Performs packet routing across gateways, intermediate hosts.
- Transport Layer
  - TCP makes conns. on network w/ sockets, using IP/ports.
  - Keeps track of order of packet delivery.
  - Ensures all data arrives at destination in order (ACK-/SYN).
  - Two basic protocols: TCP and UDP.
  - TCP is transmission control protocol, which is stream-oriented. Unduplicated and reliable.
  - UDP is user datagram protocol which gives no guarantee of delivery or duplication. (more efficient)
- Presentation Layer: converts local representations of data into its canonical form.
- Application layer: provides network services to end users, FTP clients, telnet, SMTP.

5.4 Domain Name System

- Translates symbolic hostnames into IP addresses
- Hierarchical, distributed naming system for things on the internet
- IP uses 32-bit address (4 sets of #s, 0-255)
- Each (sub)domain has 1+ authoratative DNS servers that public info to name servers.
- DNS server maintains list of resolutions.

5.5 Sockets

- BSD sockets enable communication between client/server.
- Semantics resemble pipes (files), bidirectional.
- Once one is created it can be bound to a port.
- A server assigns an address to its socket + tells all potential clients.
- Servers are passive and always waiting for clients to do something.
- A client obtains correct socket address of any server.
- Clients are active and run automatically deciding when to use a server.
- Low level ports reserved for OS.
- Each port can be bound to an address and used by an application.

6 Distributed Processing

6.1 Applications

- Databases:
  - Databases is a very common family of distrib. proc w/ client/servers.
  - The server in a database maintains it
  - Clients use transactions to interact, usually over IP. Many clients can coexist.
- Thinclients (dumb terminals / VM)
- Three tier models:
  - Application software distributed on all 3.
  - Users use a thin client, while the backend has "legacy applications"
  - Middle tier has gateway, protocol conversion, mapping, and is both a client and a server

6.2 Issues

- Lack of standards leads to middleware.
  - Sets of tools to provide uniform style of access and usage across a platform.
  - Provides standard programming interfaces/protocols to sit in the middle of client/servers.
  - Handles complexities and disparities.
  - SOA, service oriented architecture. Services with well defined interfaces are given to other groups to maintain.
  - XML via HTTP is a popular interface for communication between services.
  - RPC is another standard (remote procedure calls).
- Message passing schemes
  - Guarantees delivery if possible
  - Send the message out without reporting failure/success reduces overhead and allows queueing.
  - Blocking: send does not return control to sending process until sent, does not return control until ACK'd, or not returned until buffered for send.
  - Nonblocking: process is not suspended as result of send/received, difficult to debug but efficient.

6.3 Clusters

- Alternative to SMP— group of interconnected comps working as unified source, acts as once machine.
- SMP is easier to manage and configure.
- SMP takes up less space, uses less power.
- Clusters are better for incremental / abs scalability
- $\wedge$ superior in terms of availability
- $\wedge$ Better price/performance.
- Load balancing is a problem for clusters.
- Failure management is another issue— failure tolerant or highly available?

7 Glossary

7.1 Exam 1

- Address Translator – A functional unit that transforms virtual addresses to real addresses
- Busy waiting – the repeated execution of a loop of code while waiting for an event to occur
- Context Switch – an operation that switches the processor from one process to another, by saving all the process control block, registers, and other information for the first and replacing them with the process information for the second
- Direct Memory Access (DMA) – a form of I/O in which a special module, called a DMA module, controls the exchange of data between main memory and an I/O device. The processor sends a request for the transfer block of data to the DMA module and is interrupted only after the entire block has been transferred.
- File Allocation Table (FAT) – a table that indicates the physical location on secondary storage of the space allocated to a file. There is one file allocation table for each file.
- Job Control language (JCL) – a problem-oriented language that is designed to express statements in a job that are used to identify the job or to describe its requirements to an operating system
- Kernel – a portion of the operating system that includes the most heavily used portions of software. Generally, the kernel is maintained permanently in main memory. The kernel runs in a privileged mode and responds to calls from processes and interrupts from devices.
- Race Condition – an undesirable situation that occurs when a device or system attempts to perform two or more operations at the same time, but because of the nature of the device or system, the operations must be done in the proper sequence in order to be done correctly.
- Starvation: process is delayed forever as other processes are always given preference.
- Time Sharing – the concurrent use of a device by a number of users
- Time Slicing – a mode of operation in which two or more processes are assigned quanta of time on the same processor

7.2 Exam 2

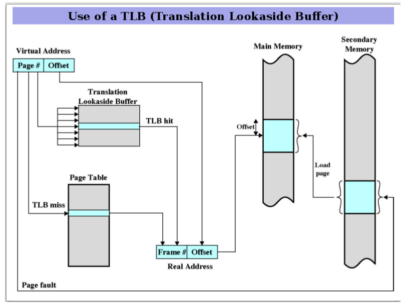
- Base Address: an address that is used as the origin in the calculation of addresses in the execution of a computer program.
- Dynamic Relocation: a process that assigns new absolute addresses to a computer program during execution so that the program may be executed from a different area of main storage.
- Indexed Sequential Access: pertaining to the organization and accessing of the records of a storage structure through an index of the keys that are stored in arbitrarily partitioned sequential files.
- Logical Address: a reference to a memory location independent of the current assignment of data to memory. A translation must be made to a physical address before the memory access can be achieved.
- Page: in virtual storage, a fixed length block that has a virtual address and that is transferred as a unit between main memory and secondary memory.
- Paging: the transfer of pages between main memory and secondary memory.

- Physical Address: the absolute location of a unit of data in memory (e.g., word or byte in main memory, block on secondary memory).
- Sequential Access: the capability to enter data into a storage device or a data medium in the same sequence as the data are ordered or to obtain data in the same order as they were entered.
- Sequential File: a file in which records are ordered according to the values of one or more key fields and processed in the same sequence from the beginning of the file.
- Spooling: the use of secondary memory as buffer storage to reduce processing delays when transferring data between peripheral equipment and the processors of a computer.
- Trojan Horse: secret undocumented routine embedded within a useful program. Execution of the program results in execution of the secret routine.
- Virtual Address: the address of a storage location in virtual storage.

8 QA Pool

8.1 Memory Management

- Page fault trap interrupt is created when a desired page frame isn't in RAM.
- Valid bit is set in the page table when it is not in RAM. This is how it knows it isn't.
- Dirty bits are set when a page frame has been modified.
- Locality: process executes in clustered pages (low fragmentation). Bad is opposite.
- Global allocation: allows a proc. to select a replacement frame from a set of all frames, even if it's given to some other process; processes can steal frames.
- Working set allocation: assumes processes execute in localities. Each process should be alloc'd enough frames for current set.
- Which of the two above are more affected by bad locality? Working set as those with bad locality have poorly defined working sets, so more page faults.
- Page fault timeline:
  - OS blocks proc, puts in waiting queue
  - Ready queue proc. is selected to run
  - DMA is initiated to load faulted page
  - Page replacement strat. ran
  - Page table updated to reflect change
- TRUE: When a DMA takes place, processor does other things.
- FALSE: DMA interrupts CPU by stealing cycles.
- Largest program that can execute on a comp. with 24-bit virtual addresses is a 2^{24} byte program.
- Same question as $\wedge$ but with hardware: can't tell, need to know virt. size.
- Address in a TLB entry PTE is physical.
- Flags in a PTE: valid bit, reference bit, dirty bit.
- Hit-ratio: in 2 level memory system (RAM-HD/cache-RAM), it is the frac of mem accesses found in master (first).
- FALSE: Misses in the TLB guarantee entries in the page table entry— X may not be resident in RAM.
- TRUE: It is possible that page tables are stored in virtual memory.
- TRUE: page sizes must be large enough to offset high cost of page faults.
- TRUE: in virt. mem. system, cannot run program whose size $>$ main mem.



- Valid bit states whether the page table entry has the disk address.
- Demand Paging: only load virtual pages as accessed.
- Prepaging: bring more pages in than needed, ones that follow.

8.2 Sockets

- The second argument to listen(sock,#) determines wait queue size, not max.
- Fat client models DO take advantage of desktop power (they ARE desktops).
- All of the following socket commands descriptors return -1 on failure.
- socket(..., SOCKET_STREAM, 0). Creates TCP socket and rets. descriptor.
- bind(sd, -struct cast madness-, len). Binds the definition of a socket to a port #.
- socket(..., SOCKET_SOCK_DGRAM, 0). Creates a UDP socket and rets. descriptor.
- accept(sd, -cast madness-, len). Blocks until client connection received, returns descriptor when it happens.

8.3 Scheduling and Processing

- Direct goals of proc. schedulers: improve response time, throughput, TAT, efficiency

8.4 Security and Triumph of the Nerds

- Most antivirus software uses emulation / heuristics.
- Logic bombs, trojan horses, and viruses require a host program to operate.
- "Bots" attack require: attack software, many vulnerable machines, and locating those machines.
- Dennis Richie and Ken Thompson basically made C/Unix. Worship them.
- Bill Gates and Paul Allen started MS in 1975.
- Xerox/PARC made drop down menus, the mouse, windows, etc.
- Steve Jobs and Steve Wozniac co-founded Apple. The former then started NeXT and was the CEO of pixar.
- MSDOS was mostly QDOS, which Tim Patterson wrote, owned by CL Computer Productions, cloned by CPM, which was written by Gary Kildall.
- Jobs saw a GUI at PARC that inspired him to computer real good.
- $\wedge$ also saw OOP and e-mail which he ignored
- $\wedge$ created Lisa after $\wedge$ which flopped. Then created Macintosh (2nd).
- BASIC language interpreter kickstarted MS into the micro-comp. bidness.
- MS got into the OS market when Kildall didn't persue IBM when they wanted a new OS. His wife/attourny didn't want to sign an NDA. Gates saw this and jumped in.
- Apple purchased NeXT and its OS NExTStep in 1996.
- Killer App: something so useful people buy comps to run it.
- Apple II's $\wedge$ was Visicalc.
- IBM's PC's $\wedge$ was Lotus 1-2-3.
- Macintosh's $\wedge$ was a WYSIWYG desktop publisher.
- IBM didn't create their own OS on their first PC as they wanted to make it super fast, under a year. They couldn't design much, just had to slap it together.
- Compaq had to reverse engineer ROM-BIOS from IBM's first PC as it was proprietary.
- IBM decided on an open architecture to save time. They bought computer components off the shelf and assembled them, hence "open architecture". As a result IBM also had to buy the OS and other software from other vendors.
- Ed Roberts at MITS built the first commercially available PC in 1975.
- Gordon Moore is an Intel founder.
- Altair 8800 was the world's first personal computer and was designed by Ed Roberts in 1975.
- First mass market PC company is Apple.

8.5 Weird Questions

- RAID #'s: 1, mirror; 0, merge drives (splits data even); 10|2, striping, error detection, and fault tolerance (common).

Part II

Mathematics

Chapter 3

MATH 1214 - Calculus I

Calculus I Master

Course: `../math1214-calculus-i/master.pdf`

Document 8

Included from: `../math1214-calculus-i/master.pdf`

Calculus I: Single-Variable Calculus

Illya Starikov

June 30, 2025

Contents

1 Functions

1.1 Review of Functions

A **function** is a rule that assigns to each value x in a set D a unique value denoted $f(x)$. The set D is the **domain** of the function. The **range** is the set of all values of $f(x)$ produced as x varies over the domain.

Vertical Line Test

A graph represents a function if and only if it passes the **vertical line test**. Every vertical line intersects the graph at most once. A graph that fails this test does not represent a function.

Composite Functions

Given two functions f and g, the composite functions $f \circ g$ is defined by $(f \circ g)(x) = f(g(x))$. It is evaluated in two steps: $y = f(u)$, where $u = g(x)$. The domain of $f \circ g$ consists of all x in the domain of g such that $u = g(x)$ is in the domain of f.

Symmetry in Functions

An **even function** has the property that $f(-x) = f(x)$, for all x in the domain. The graph of an even function symmetric about the y-axis. Polynomials consisting of only even powers of the variable (of the form x^{2n}, where n is a nonnegative integer) are even functions.
An **odd function** f has the property that $f(-x) = -f(x)$, for all x in the domain. The graph of an odd function is symmetric about the origin. Polynomials consisting of only odd powers of the variable (of the form x^{2n+1}, where n is a nonnegative integer) are odd functions.

1.2 Review of Functions

A **function** is a rule that assigns to each value x in a set D a unique value denoted $f(x)$. The set D is the **domain** of the function. The **range** is the set of all values of $f(x)$ produced as x varies over the domain.

Vertical Line Test

A graph represents a function if and only if it passes the **vertical line test**. Every vertical line intersects the graph at most once. A graph that fails this test does not represent a function.

Composite Functions

Given two functions f and g, the composite functions $f \circ g$ is defined by $(f \circ g)(x) = f(g(x))$. It is evaluated in two steps: $y = f(u)$, where $u = g(x)$. The domain of $f \circ g$ consists of all x in the domain of g such that $u = g(x)$ is in the domain of f.

Symmetry in Functions

An **even function** has the property that $f(-x) = f(x)$, for all x in the domain. The graph of an even function symmetric about the y-axis. Polynomials consisting of only even powers of the variable (of the form x^{2n}, where n is a nonnegative integer) are even functions.
An **odd function** f has the property that $f(-x) = -f(x)$, for all x in the domain. The graph of an odd function is symmetric about the origin. Polynomials consisting of only odd powers of the variable (of the form x^{2n+1}, where n is a nonnegative integer) are odd functions.

1.3 Representing Functions

Some brief families of functions can include

Polynomials are functions of the form

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0,$$

where the **coefficients** $a_0, a_1, \ldots, a_n$ are real numbers with $a_n \neq 0$ and the nonnegative integer n is the **degree** of the polynomial. The domain of any polynomial is the set of all real numbers. An nth-degree polynomial can have as many as n real **zeros** or **roots** — values of x.

Rational Functions are ratios of the form $f(x) = p(x)/q(x)$, where p and q are polynomials. Because division by zero is prohibited, the domain of a rational function is the set of all real numbers except those for which the denominator is zero.

Algebriac Functions are constructed using the operations of algebra: addition, subtraction, multiplication, division, and roots. Examples of algebraic functions are $f(x) = \sqrt{2x^3 + 4}$ and $f(x) = x^{1/4}(x^3 + 2)$. In general, if an even root (square root, fourth root, and so forth) appears, then the domain does not contain points at which the quantity under the root is negative (and perhaps other points).

Exponential Functions have the form $f(x) = b^x$, where the base $b \neq 1$ is a positive real number. Closely associated with exponential functions are logarithmic functions of the form $f(x) = \log_b x$, where $b > 0$ and $b \neq q$. An exponential function has a domain consisting of all real numbers. Logarithmic functions are defined for positive, real numbers. The most important function is the **natural exponential function** $f(x) = e^x$, with base $b = e$, where $e \approx 2.71828\ldots$ is one of the fundamental constants of mathematics. Associated with the natural exponential function is the **natural logarithmic function** $f(x) = \ln x$, which also has the base $b = e$.

Trigonometric Functions are $\sin x$, $\cos x$, $\tan x$, $\sec x$, and $\cos x$; they are fundamental to mathematics and many areas of application. Also important are their relatives, the **inverse trigonometric functions**.

Transcendental Functions Trigonometric, exponential, and logarithmic functions are few examples of a large family called transcendental functions.

Transformations

Given the real numbers a, b, c, and d and the function f, the graph of $y = cf(a(x - b)) + d$ is obtained from the graph of $y = f(x)$ in the following steps.

$$
\begin{aligned}
y = f(x) &\xrightarrow{\text{horizontal scaling by a factor of } |a|} y = f(ax) \\
&\xrightarrow{\text{horizontal shift by } b \text{ units}} y = f(a(x-b)) \\
&\xrightarrow{\text{vertical scaling by a factor of } |c|} y = cf(a(x-b)) \\
&\xrightarrow{\text{horizontal scaling by a factor of |a|}} y = cf(a(x-b)) + d
\end{aligned}
$$

1.4 Inverse, Exponential, and Logarithmic Functions

The Natural Exponential Function

The **natural exponential function** is $f(x) = e^x$, which as the base $e = 2.718281828459\ldots$.

Inverse Function

Given a function f, its inverse (if it exists) is a function f^{-1} such that whenever $y = f(x)$, then $f^{-1}(y) = x$.

One-to-One Functions and the Horizontal Line Test

A function f is **one-to-one** on a domain D if each value of $f(x)$ corresponds to exactly one value of x in D. More precisely, f is one-to-one on D is $y(x_1) \neq f(x_2)$ whenever $x_1 \neq x_2$ for x_1 and x_2 in D. The **horizontal line test** says that every horizontal line intersects the graph of a one-to-one function at most once.

Existence of Inverse Functions

Let f be one-to-one function on a domain D with a range R. Then f has a unique inverse f^{-1} with domain R and range D such that

$$f^{-1}(f(x)) = x \qquad \text{and} \qquad f(f^{-1}(y)) = y,$$

where x is in D and y is in R.

Finding an Inverse Function

Suppose f is one-to-one on an interval I. To find f^{-1}:

- Solve $y = f(x)$ for x. If necessary, choose the function that corresponds to I.
- Interchange x and y and write $y = f^{-1}(x)$.

Logarithmic Function Base b

For any base $b > 0$, with $b \neq 1$, the **logarithmic function base** b, denoted $y = \log_b x$, is the inverse of the exponential function $y = b^x$. The inverse of the natural exponential function with base $b = e$ is the **natural logarithm function**, denoted $y = \ln x$.

Inverse Relations For Exponential and Logarithmic Functions

For any base $b > 0$, with $b \neq 1$, the following inverse relations hold:

- $b^{\log_b x} = x$, for $x > 0$
- $\log_b b^x = x$, for any real values of x

Change-of-Base Rules

Let b be a positive real number with $b \neq 1$. Then

$$b^x = e^{x \ln b}, \text{for all } x \qquad \text{and} \qquad \log_b x = \frac{\ln x}{\ln b}, \text{for } x > 0$$

More generally, if c is a positive real number with $c \neq 1$, then

$$b^x = c^{x \log_c b}, \text{for all } x \qquad \text{and} \qquad \log_b x = \frac{\log_c x}{\log_c b}, \text{for } x > 0$$

1.5 Trigonometric Functions and Their Inverses

Let $P(x,\, y)$ be a point on a circle of radius r associated with the angle θ. Then

$$\sin\theta = \frac{y}{r} \qquad \cos\theta = \frac{x}{r} \qquad \tan\theta = \frac{y}{x} \tag{1}$$

$$\cot\theta = \frac{x}{y} \qquad \sec\theta = \frac{r}{x} \qquad \csc\theta = \frac{r}{y} \tag{2}$$

$$\tag{3}$$

Trigonometric Identities

Reciprocal Identities

$$\tan\theta = \frac{\sin\theta}{\cos\theta} \qquad \cot\theta = \frac{1}{\tan\theta} = \frac{\cos\theta}{\sin\theta} \tag{4}$$

$$\csc\theta = \frac{1}{\sin\theta} \qquad \sec\theta = \frac{1}{\cos\theta} \tag{5}$$

Pythagorean

$$\sin^2\theta + \cos^2\theta = 1 \qquad 1 + \cot^2\theta = \csc^2\theta \qquad \tan^2\theta + 1 = \sec^2\theta \tag{6}$$

Double- and Half-Angle Formulas

$$\sin 2\theta = 2\sin\theta\cos\theta \qquad \cos 2\theta = \sin^2\theta - \sin^2\theta \tag{7}$$

$$\cos^2\theta = \frac{1+\cos 2\theta}{2} \qquad \sin^2\theta = \frac{1-\cos 2\theta}{2} \tag{8}$$

Period of Trigonometric Function

The function $\sin\theta$, $\cos\theta$, $\sec\theta$, and $\csc\theta$ have a period of 2π

$$\sin(\theta + 2\pi) = \sin\theta \qquad \cos(\theta + 2\pi) = \cos\theta \tag{9}$$

$$\sec(\theta + 2\pi) = \sec\theta \qquad \csc(\theta + 2\pi) = \csc\theta \tag{10}$$

for all θ in the domain.
The functions $\tan\theta$ and $\cot\theta$ have a period of π:

$$\tan(\theta + \pi) = \tan\theta \qquad \cot(\theta + \pi) = \cot\theta \tag{11}$$

for all θ in the domain.

Inverse Sine and Cosine

$y = \sin^{-1} x$ is the value of y such that $x = \sin y$, where $-\frac{\pi}{2} \leq y \leq \frac{\pi}{2}$. $y = \cos^{-1} x$ is the value of y such that $x = \cos y$, where $0 \leq y \leq \pi$. The domain of both $\sin^{-1} x$ and $\cos^{-1} x$ is $\{x : -1 \leq x \leq 1\}$.

Other Inverse Trigonometric Functions

- $\tan^{-1} x$ is the value of y such that $x = \tan y$, where $-\frac{\pi}{2} < \frac{\pi}{2}$.
- $\cot^{-1} x$ is the value of y such that $x = \tan y$, where $0 < y < \pi$.

The domain of both $\tan^{-1} x$ and $\cot^{-1} x$ is $\{x : -\infty < x < \infty\}$

- $\sec^{-1} x$ is the value of y such that $x = \sec y$, where $0 < y < \pi$ with $y \neq \frac{\pi}{2}$
- $\tan^{-1} x$ is the value of y such that $x = \tan y$, where $-\frac{\pi}{2} < \frac{\pi}{2}$.

2 Limits

2.2 Definitions of Limits

Limits of a Function (Preliminary)

Suppose the function f is defined for all x near a except possibly at a. If $f(x)$ is arbitrarily close to L (as close to L as we like) for all x sufficiently close (but not equal) to a, we write

$$\lim_{x \to a} f(x) = L$$

and say the limit of $f(x)$ as x approaches a equals L.

One-Sided Limits

1 Right-sided limits Suppose f is defined for all x near a with $x > a$. If $f(x)$ is arbitrarily close to L for all x sufficiently close to a with $x > a$, we write

$$\lim_{x \to a^+} f(x) = L \tag{12}$$

and say the limit of $f(x)$ as x approaches a from the right equals L.

2 Left-sided limits Suppose f is defined for all x near a with $x < a$. If $f(x)$ is arbitrarily close to L for all x sufficiently close to a with $x < a$, we write

$$\lim_{x\to a^-} f(x) = L \tag{13}$$

and say the limit of $f(x)$ as x approaches a from the left equals L.

Relationship Between One-Sided and Two-Sided Limits

Assume f is defined for all x near a except possibly at a. Then $\lim\limits_{x\to a} f(x) = L$ if and only if $\lim\limits_{x\to a^+} = \lim\limits_{x\to a^-} = L$.

2.3 Techniques For Computing Limits

Limits of Linear Functions

Let a, b, and m be real numbers. For Linear functions $f(x) = mx + b$,

$$\lim_{x\to a} f(x) = f(a) = ma + b \tag{14}$$

Limit Laws

Assume $\lim\limits_{x\to a} f(x)$ and $\lim\limits_{x\to a} g(x)$ exist. The following properties hold, where c is a real number, and $m > 0$ and $n > 0$ are integers.

1 Sum $\lim\limits_{x\to a} [f(x) + g(x)] = \lim\limits_{x\to a} f(x) + \lim\limits_{x\to a} g(x)$

2 Difference $\lim\limits_{x\to a} [f(x) - g(x)] = \lim\limits_{x\to a} f(x) - \lim\limits_{x\to a} g(x)$

3 Constant Multiple $\lim\limits_{x\to a} [c\, f(x)] = c \lim\limits_{x\to a}$

4 Product $\lim\limits_{x\to a} [f(x)g(x)] = \left[\lim\limits_{x\to a} f(x)\right] \left[\lim\limits_{x\to a} g(x)\right]$

5 Quotient $\lim\limits_{x\to a} \left[\frac{f(x)}{g(x)}\right] = \frac{\lim\limits_{x\to a} f(x)}{\lim\limits_{x\to a} g(x)}$, provided $\lim\limits_{x\to a} g(x) \neq 0$

6 Power $\lim\limits_{x\to a} [f(x)]^n = \left[\lim\limits_{x\to a} f(x)\right]^n$

7 Fractional Power $\lim\limits_{x\to a} [f(x)]^{n/m} = \left[\lim\limits_{x\to a} f(x)\right]^{n/m}$, provided $f(x) \geq 0$, for x near a, if m is even and n/m is reduced to lowest terms.

Limits of Polynomial and Rational Functions

Assume p and q are polynomials and a is a constant

- Polynomial functions: $\lim_{x\to a} p(x) = p(a)$
- Rational functions: $\lim_{x\to a} \frac{p(x)}{q(x)} = \frac{p(a)}{q(a)}$, provided $q(a) \neq 0$

Limit Laws For One-Sided Limits

Laws 1–6 hold with $\lim_{x\to a}$ replaced by $\lim_{x\to a^+}$ or $\lim_{x\to a^-}$. Law 7 is modified as follows, assume $m > 0$ and $n > 0$ are integers.

7 Fractional Power

- $\lim_{x\to a^+} [f(x)]^{n/m}$, provided $f(x) \geq 0$, for x near a with $x > a$, if m is even and n/m is reduced to lowest terms
- $\lim_{x\to a^-} [f(x)]^{n/m}$, provided $f(x) \geq 0$, for x near a with $x < a$, if m is even and n/m is reduced to lowest terms

The Squeeze Theorem

Assume the function f, g, and h satisfy $f(x) \leq g(x) \leq h(x)$, for all values of x near a, except possibly at a. If $\lim_{x\to a} f(x) = \lim_{x\to a} h(x) = L$, then $\lim_{x\to a} g(x) = L$.

2.4 Infinite Limits

Suppose f is defined for all x near a. If $f(x)$ grows arbitrarily large for all x sufficiently close (but not equal) to a, we write

$$\lim_{x\to a} f(x) = \infty \tag{15}$$

We say the limit of $f(x)$ as x approaches a is infinity.
If $f(x)$ is negative and grows arbitrarily large in magnitude for all x sufficiently close (but not equal) to a, we write

$$\lim_{x\to a} f(x) = -\infty \tag{16}$$

In this case, we say the limit of $f(x)$ as x approaches a is negative infinity. In both cases, *the limit does not exist.*

One-Sided Infinite Limits

Suppose f is defined for all x near a with $x > a$. If $f(x)$ becomes arbitrarily large for all x sufficiently close to a with $x > a$, we write $\lim_{x \to a^+} f(x) = \infty$. The one-sided infinite limit $\lim_{x \to a^+} = -\infty$, $\lim_{x \to a^-} f(x) = \infty$, and $\lim_{x \to a^-} = -\infty$ are defined analogously.

Vertical Asymptote

If $\lim_{x \to a} f(x) = \pm\infty$, $\lim_{x \to a^+} = \pm\infty$ or $\lim_{x \to a^-} f(x) = \pm\infty$, the line $x = a$ is called a **vertical asymptote** of f.

2.5 Limits at Infinity

If $f(x)$ becomes arbitrarily close to a finite number L for all sufficiently large and positive x, then we write

$$\lim_{x \to \infty} f(x) = L \tag{17}$$

We say the limit of $f(x)$ as x approaches infinity is L. In this case the line $y = L$ is a **horizontal asymptote** of f. The limit at negative infinity, $\lim_{x \to -\infty} f(x) = M$, is defined analogously. When the limit exists, the horizontal asymptote is $y = M$.

Infinite Limits at Infinity

If $f(x)$ becomes arbitrarily large as x becomes arbitrary large, then we write

$$\lim_{x \to \infty} f(x) = \infty \tag{18}$$

The limits $\lim_{x \to \infty} = -\infty$, $\lim_{x \to -\infty} = \infty$, and $\lim_{x \to -\infty} = -\infty$ are defined similarly.

Limit of Infinity at Powers and Polynomials

Let n be a positive integer and let p be the polynomial $p(x) = a_n x^m + a_{n-1}x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$, where $a_n \neq 0$.

1. $\lim_{x \to \pm\infty} x^n = \infty$ when n is even.
2. $\lim_{x \to \infty} x^n = \infty$ and $\lim_{x \to -\infty} x^n = -\infty$ when n is odd.

3. $\lim_{x\to\pm\infty} \frac{1}{x^n} = \lim_{x\to\pm\infty} x^{-n} = 0$

4. $\lim_{x\to\pm\infty} p(x) = \infty or - \infty$, depending on the degree of the polynomial or the leading coefficient of a_n.

End Behavior and Asymptotes of Rational Functions

Suppose $f(x) = \frac{p(x)}{q(x)}$ is a rational function, where

$$p(x) = a_m x^m + a_{m-1} + \cdots + a_2 x^2 + a_1 x + a_0 \tag{19}$$

and

$$q(x) = b_m x^m + b_{m-1} + \cdots + b_2 x^2 + b_1 x + b_0 \tag{20}$$

with $a_m \neq 0$ and $b_n \neq 0$.

1. If $m < n$ then $\lim_{x\to\pm\infty} f(x) = 0$, and $y = 0$ is a horizontal asymptote of f.

2. If $m = n$, then $\lim_{x\to\pm\infty} f(x) = a_m/b_n$, and $y = a_m/b_n$ is a horizontal asymptote.

3. If $m > n$, then $\lim_{x\to\pm\infty} f(x) = \infty$ or $-\infty$, and f has no horizontal asymptote.

4. If $m = n + 1$, then $\lim_{x\to\pm\infty} f(x) = \infty$ or $-\infty$, f has no horizontal asymptote, but f has a slant asymptote.

5. Assuming that $f(x)$ is in reduced form (p and q share no common factors), vertical asymptotes occur at the zeros of q.

End Behavior of e^x, e^{-x}, and $\ln x$

The end behavior for e^x and e^{-x} on $(-\infty, \infty)$ and $\ln x$ on $(0, \infty)$ is given by the following limits:

$$\lim_{x\to\infty} e^x = \infty \qquad \text{and} \qquad \lim_{x\to-\infty} e^x = 0 \tag{21}$$

$$\lim_{x\to\infty} e^{-x} = \infty \qquad \text{and} \qquad \lim_{x\to-\infty} e^x = 0 \tag{22}$$

$$\lim_{x\to 0^+} \ln x = -\infty \qquad \text{and} \qquad \lim_{x\to\infty} \ln x = \infty \tag{23}$$

Part III

Engineering Sciences

Chapter 4

PHYS 1135 - Physics I

Interference

Course: `../phys1135-physics-i/Intereferance.pdf`

Document 9

Included from: `../phys1135-physics-i/Intereferance.pdf`

1 Interference

$$\begin{aligned} & sin(kx - wt) + sin(kx + wt) \\ = \quad & 2sin\frac{2kx}{2}cos\frac{2\omega t}{2} \\ = \quad & sin(\omega_1 t) + sin(\omega_2 t) \\ = \quad & 2sin(\frac{\omega_1 + \omega_2}{2}t) + 2cos(\frac{\omega_1 - \omega_2}{2}t) \end{aligned}$$

$$\begin{aligned} sin(a) + sin(b) &= 2sin\frac{a+b}{2}cos\frac{a-b}{2} \\ cos(a) + cos(b) &= 2cos\frac{a+b}{2}cos\frac{a-b}{2} \end{aligned}$$

- This is a standing wave.
- Where is the second wave coming from? The reflection

Chapter 5

PHYS 2305 - Modern Physics

Modern Physics Review

Course: `../phys2305-modern-physics/review.pdf`

Document 10

Included from: `../phys2305-modern-physics/review.pdf`

Modern Physics Review

Illya Starikov

June 30, 2025

1 Special Relativity

From relativity, we know

1. All inertial reference frames are equivalent.
2. The speed of light is the same in all inertial reference frames.

From this, we notice that

1. Time and space depend on velocity ($\vec{v}$).
2. Moving clocks appear to run slow.
3. The length of a moving object, in the direction of motion, will appear shorter.

Eloquently, this can be described as

$$\boxed{t = \gamma t_0} \qquad \boxed{L = \frac{L_0}{\gamma}}$$

Where $\boxed{\gamma = \frac{1}{\sqrt{1 - v^2/c^2}}}$, t is the time according to the outside observer, t_0 to be proper time (i.e. the time measured by the moving object), L is length to outside observer, and L_0 is proper length.

2 Energy and Momentum

Relativistic energy and momentum can be defined as

$$E = \gamma m_0 c^2 \qquad \vec{p} = \gamma m_0 \vec{v}$$

From the this, we can derive the more fundamental equation:

$$E^2 = m_0^2 c^4 + p^2 c^2 \tag{1}$$

For kinetic energy, we can eloquently describe it as $(\gamma - 1)m_0 c^2$, this is simply the rest mass energy (mc^2) subtracted from the total energy (γmc^2). At low speeds (i.e. $v \ll c$), we can simply use a Taylor Series expansion to get $E \approx {}^1/_2 m_0 v^2 + \frac{3m_0 v^4}{8c^2} + \cdots$. Ignoring the other terms, we get $E \approx {}^1/_2 mv^2$, classical energy!
If we take m_0 to be 0, this implies $E = pc$ (From Equation 1). Using de Broglie wavelength ($\lambda = {}^h/_p$), this is implies $E = h\nu$. This can be combined with the fact that $\lambda\nu = c$, which allows many permutations of the equations.
We can also show that the kinetic energy $KE = \frac{p^2}{2m}$ To summarize, for $m_0 = 0$,

$$E = h\nu = \frac{hc}{\lambda} = pc$$

3 The Three Experiments

There were three experiments done to prove the quantum nature of light and particles.

3.1 The Photoelectric Effect

Suppose we have a sea of electrons within a metal. It takes some work to escape the sea, described by the work function Φ. We can relate the

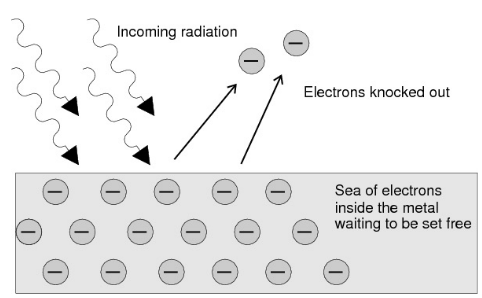


Figure 1 – The Photoelectric Effect.

energy of the photon, the work function, and the resulting energy of the electrons by $h\nu = \Phi + KE_{\max}$.

3.2 Compton Scattering

We can scatter a photon off an electron, inelastically, and after some tedious math we can realize $\lambda_2 - \lambda_1 = \frac{h}{m_0 c}(1 - \cos\theta)$, where λ_2 is the wavelength after scattering, λ_1 is the wavelength before the scattering, and m_0 the electron rest mass. From this we realize that there is a decrease in the energy of the photon, resulting in an increase of wavelength, which we know as the Compton effect.

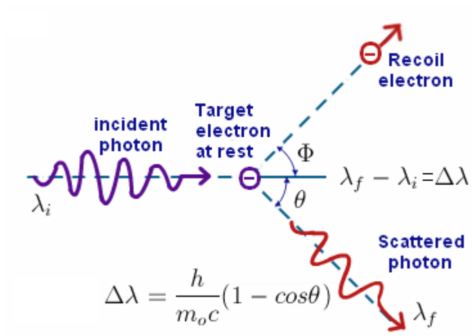


Figure 2 – Compton scattering.

3.3 Blackbody Radiation

Blackbody radiation refers to an object or system which absorbs all radiation incident upon it and re-radiates it. This effect can be characterized by the radiating system alone; it does not depend on the type of radiation incident upon it. The radiated energy can be considered to be produced by standing wave or resonant modes of the cavity which is radiating. The major effect of this is that the modes must be quantized. We can define the energy per unit volume per unit frequency

$$U(\nu) = \frac{8\pi\nu^2}{c^3} \frac{h\nu}{e^{\frac{h\nu}{kT}} - 1}$$

where k is the Boltzmann constant and T is the absolute temperature of the body. The entity $\frac{h\nu}{e^{h\nu/kT}-1}$ is the average energy per mode, and $\frac{8\pi\nu^2}{c^3}$ counts the number of modes available.

4 Wave Nature of Massive Particles

$$|\psi(x)|^2 dx = \text{the probability of finding the particle in the range of } x \text{ to } x + dx$$

Because $|\psi(x)|^2$ is a continuous probability distribution, we must normalize the wavefunction so that the probability has to add up to 100%,

$$\int_{-\infty}^{\infty} |\psi(x)|^2 \, dx = 1$$

4.1 Uncertainty Principle

It can be derived that for an particular measurement,

$$\Delta t \Delta E \geq \frac{\hbar}{2} \qquad \hbar = \frac{h}{2\pi}$$

Or, more famously,

$$\Delta x \Delta p \geq \frac{\hbar}{2} \qquad \hbar = \frac{h}{2\pi}$$

4.2 Infinite Potential Well

Imagine a potential wall where the wall go to infinity, with a distance L between the walls. Our wave function ψ must have the boundary conditions $\psi(x = 0) = 0$ and $\psi(x = L) = 0$. From this we realize we can only fit half-wavelengths of the wave into the box; that is, $n\lambda/2 = L$. To calculate the energy,

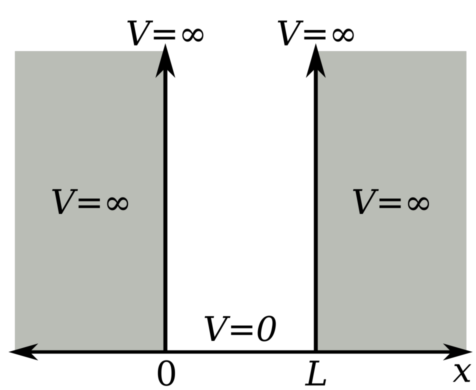


Figure 3 – Particle in a box (or infinite potential well).

$$\boxed{E = \frac{p^2}{2m} = \frac{n^2 h^2}{8mL^2}}$$

because $\lambda = \frac{2L}{n}$ and $p = \frac{h}{\lambda}$.

4.3 Wave Motion

We know we can describe just about any wave by $\cos(kx - \omega t)$, where $k = \frac{2\pi}{\lambda}$ and $\omega = 2\pi\nu$. Consequently, $p = {}^h/_\lambda = \hbar k$ and $E = h\nu = \hbar\omega$.
Furthering our wave mathematics, we can consider the phase velocity as

$$v_p = \frac{\omega}{k} = \frac{E}{p} = \nu\lambda$$

When considering a group, we can calculate the velocity of the group (or the packet), v_g, as

$$v_g = \frac{\partial\omega}{\partial k} = \frac{\partial E}{\partial p}$$

4.4 Hydrogen Atom

As we have proven with our three experiments, energy states are quantized. This implies that the energy levels in an atom come in integer levels (i.e. energy level $n = 1, 2, 3, \ldots, \infty$, where ∞ is ionization).

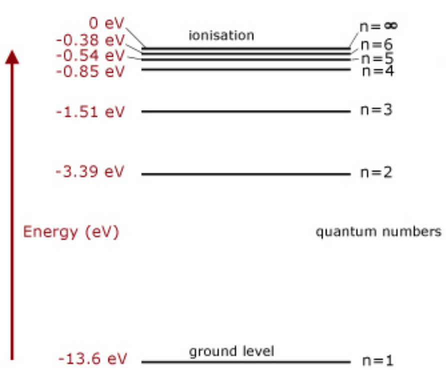


Figure 4 – Energy levels of hydrogen atom.

From this, we can determine that the energy at any level $E_n = \frac{E_1}{n^2}$, where $E_1 < 0$. For photo-absorption,

$$\begin{aligned} h\nu = E_f - E_i &= \frac{E_1}{n_f^2} - \frac{E_1}{n_i^2} \\ &= \boxed{E\left(\frac{1}{n_f^2} - \frac{1}{n_i^2}\right)} \end{aligned}$$

Similarly, for emission,

$$
\begin{aligned}
h\nu = E_i - E_f &= \frac{E_1}{n_i^2} - \frac{E_1}{n_f^2} \\
&= E\left(\tfrac{1}{n_i^2} - \tfrac{1}{n_f^2}\right)
\end{aligned}
$$

Part IV

Liberal Arts & General Education

Chapter 6

PSYC 1101 - Psychology 101

Steven Pinker Assignment

Course:
../psyc1101-psychology-101/Assignments/steven_pinker.pdf

Document 11

Included from:
../psyc1101-psychology-101/Assignments/steven_pinker.pdf

Steven Pinker: The Language of Cognition

Illya Starikov
Professor Amanda Burch
General Psychology
June 30, 2025

Illya Starikov

Professor Amanda Burch

General Psychology

June 30, 2025

Steven Pinker: The Language of Cognition

Steven Pinker is one of the most influential psychologists of our time. Aside from being named Time's 100 most influential people in 2004 and winning Foreign Policy's 100 top public intellectuals in 2004 and 2008, he has won awards from the National Academy of Sciences (which he was elected to in 2016), the Royal Institution, American Psychological Association, the American Humanist Association, and the Cognitive Neuroscience Society. Although his work is primarily in linguistics and its relation to psychology, he has also does work work in cognitive science and visual cognition. From his research he has published ten books; some which include The Language Instinct, How the Mind Works, The Blank Slate, The Stuff of Thought and, more recently, The Better Angels of Our Nature. Before diving into his formal research, it might be more helpful to look at his earlier years and their influence on his later life.

Steven Pinker was born in Montreal, Quebec, in 1954 to a Jewish family. Although his parents were devoted Jewish, he proudly renounced the faith, a sentiment that is found to this day:

> ““The Bible is a manual for rape, genocide, and the destruction of families...Religion has given us stonings, witch burnings, crusades, Inquisitions, jihads, fatwas, suicide bombers...and mothers who drown their children in the

river," he said."

This rejection of religion ultimately lead him down a life of science; specifically, psychology.

After graduating with a bachelor's degree from McGill University in 1976, he later finished his doctorate only three years later from Harvard. Before finally settling as the Johnstone Family Professor in the Department of Psychology at Harvard, where he currently works, he bounced around between Stanford, MIT, University of California, Santa Barbara and Harvard. The attendance of these prestigious colleges was not without merit, of course; Steven Pinker had accomplished quite a bit at this time.

Pinker's most notable work is not just the work of the mind, but of his enthusiasm for language. This fascination came to him in graduate school, which he decided to pursue in his research. Roughly during this time, he had published two books. His first book, Language Learnability and Language Development, was regarded as "A fiercely reasoned, bently written landmark of psychological science.". As hinted by the name, the book sheds light on the problem of language learning; in particular, language acquisition in children. In the book, he gives a brief summary on the matter:

> "The core assumption of the theory is that children are innately equipped with algorithms designed to acquire the grammatical rules and lexical entries of a human language. The algorithms are triggered at first by the meanings of the words in the input sen- tences and knowledge of what their referents are doing, gleaned from the context. Their outputs, the first grammatical rules, are used to help analyze subsequent inputs and to trigger other learning algorithms, which come in sets tailored to the major components of language. Empirically,

> children do seem to show the rapid but piecemeal acquisition of adultlike rules predicted by the theory, with occasional systematic errors that betray partly acquired rules.” (Pinker, 2)

As such, he has spent many years of his later years still studying the lexical structure of language. However, he had another passion, which he touched on in his second book, Visual Cognition: Computational Models of Cognition and Perception.

Visual Cognition addresses some of the more difficult question of human visualization, such as:

> “How do we recognize objects? How do we reason about objects when they are absent and only in memory? How do we conceptualize the three dimensions of space? Do different people do these things in different ways? And where are these abilities located in the brain?”

Throughout the remainder of the book, Pinker goes about answer these questions through past studies, modeling of cognitive processes, and new experimental techniques.

Although his main claim to fame was progressive studies in these two fields, visual cognition and language, he has done additional studies regarding violence, written additional books (regarding writing, violence, humanitarian efforts, and the human mind), advocated for science based thinking, participated in public debates, and occasionally done interviews with notable online celebrities.

In general, I do agree with Steven Pinker. His work for language, in my limited exposure to it, has always been intuitive and downright spot on. In one of his Ted Talks, What our language habits reveal, a particular statement stands out:

> I think the key idea is that language is a way of negotiating relationships, and human relationships fall into a number of types. There's an influential taxonomy by the anthropologist Alan Fiske, in which relationships can be categorized, more or less, into communality, which works on the principle "what's mine is thine, what's thine is mine," the kind of mindset that operates within a family, for example; dominance, whose principle is "don't mess with me;" reciprocity, "you scratch my back, I'll scratch yours;" and sexuality, in the immortal words of Cole Porter, "Let's do it."

This gives a more modest expression of language — instead of an overly-romanticized art, language can be a negotiation, among many things. And Pinker continues through the talk discussing these strictly-scientific and psychological definitions of language and their effects on our daily lives.

Although I do agree with his work, I do not particularly see his work around me; even if it is ever-present. Language does give a direct method of communication that I may not particularly give attention to, but it does affect my daily life.

Works Cited

Pinker, Steven A. “About.” Steven Pinker. Harvard, n.d. Web. 28 June 2016.

Pinker, Steven A. “Language Learnability and Language Development.” Steven Pinker. Steven Pinker, 17 Aug. 2015. Web. 28 June 2016.

Pinker, Steven A. “Visual Cognition: Computational Models of Cognition and Perception.” Steven Pinker. Harvard, 08 Aug. 2015. Web. 28 June 2016.

Pinker, Steven. Language Learnability and Language Development. Cambridge, MA: Harvard UP, 1984. Print.

Pinker, Steven. “Transcript of “What Our Language Habits Reveal”” Steven Pinker: What Our Language Habits Reveal. TED, Sept. 2007. Web. 28 June 2016.

Quantum Cognition

Course:
`../psyc1101-psychology-101/Assignments/quantum_cognition`

Document 12

Included from:
`../psyc1101-psychology-101/Assignments/quantum_cognition.pdf`

Illya Starikov

Professor Amanda Burch

General Psychology

June 30, 2025

Quantum Cognition: A New Theoretical Approach to Psychology

For decades, judgments and human cognition has been precedented on two different concepts — heuristic and rational. The heuristic approach “posits that, to make judgments and decisions, people tend to employ simple heuristics (e.g., representativeness, anchoring-and-adjustment, take-the-best)”, while the rational approach “posits that people can derive inferences from the Bayes rule and decisions from the expected utility rule in a rational manner”. These two methods have been proven an intuitive model; however, a third approach has emerged: quantum cognition. This framework for cognition assumes that humans abide basic axioms from probability theory and quantum physics. However, these axioms can be proven to be less intuitive than their former counterparts (heuristic and rational).

The theory is quite younger than quantum physics (quantum cognition being 20 years while quantum physics being over 100 years old), but that time discrepancy has given researches the ability to draw “inspiration from both the mathematical structure of quantum theory and its associated dynamic principles.”

As for how quantum physics relates to cognition, less us first look at an example of the disparity between the logic of quantum physics and the logic of boolean algebra. The statement A and B (more formally, $A \cap B$ denotes A and B, and $P(A \cap B)$ denotes the probability of A and B) in classical boolean algebra has the commutative property, meaning one can write $P(A \cap B)$ or $P(B \cap A)$ and there would be no difference. This is not the case in quantum physics, nor is it in cognition. In quantum physics, the ordering does matter; as it does for human cognition. Occasionally, the $\cup$ can be replaced with before (i.e. A before B).

To further clarify this, let us take two examples. First, suppose a student (name him Illya) turns in two papers, paper A and paper B. In paper A, he does an excellent job, citing the paper, pulling examples from the paper, etc. The second paper, Illya does well, but not nearly to the level of excellence as the first. In this case, the order matters: if his first paper (paper A) was extraordinary, there will be that bias for the second paper (paper B) to be done. In this instance, the probability of — $P(\cdots)$ — doing well is not commutative; $P(A \cup B) \neq P(B \cup A)$, the probability of doing well on paper A and paper B are not quite the same as doing well on paper B and A. However, this is not always the case; for example we can contemplate whether to fail Illya on the second paper while we contemplate what is for dinner, as we often might do. This provides a simple introduction to the principle of complementary, which "leads directly to the best-known principle of quantum theory — the uncertainty principle."

"The [uncertainty] principle holds that when we are certain about a quantum particle's position, we are necessarily uncertain about its momentum, and vice versa." This is important in the context of incompatible events (events that can't be considered at the same time, such as Illya's papers). When one has two events dependent on the order, they can't think of them, therefore they can't think of them at the same time; just like the uncertainty of a particle's position and momentum.

Now that we have this background knowledge, we shall consider another quantum physics approach to understanding the brain itself: vector spaces. A vector space, in mathematics, is a space of vectors; and a vector is an object with direction and magnitude. This might seem abstract, but "it is analogous to the distributed input across nodes in a connectionist neural network". With this vector space, and a projector (something that maps a space to a subspace), this can "provides algorithm-level predictions for this more complex neural implementation."

However, these are all just new applications of quantum theory to cognition — we can "highlight the expressive power of quantum models by pointing out that they have already been applied to a broad range of cognitive phenomena, including perception, memory, conceptual combinations, attitudes, probability judgments, causal reasoning, decision making, and strategic games". This is still a maturing field, as we still do not quite fully understand all of quantum physics, but applications are being found and used everyday.

In my personal opinion, I believe the article is controversial. If psychology does a paradigm shift where quantum physics is the framework to understand human cognition, this particular field might not be quite as fruitful. Can you imagine a third-year psychology major walking into class and seeing $-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2} + V\psi = E\psi$ (time–independent Schrödinger's equation) for the first time? Quantum cognition is based on quantum mechanics, and quantum mechanics is based on heavy math (differential equations, complex linear algebra, multi-dimensional calculus) and every kind of physics imaginable (Newtonian mechanics, electromagnetism, theoretical, etc.). Even taking the most primitive example of physics in this article, the uncertainty principle is understood through analogy, not physics. The moving parts of this problem are: the principle itself ($\sigma_x\sigma_p \geq \frac{\hbar}{2}$), momentum ($\vec{p} = m\vec{v}$), and position (usually denoted by the vector $\vec{r}$). These are hard, mathematical and physics problems to understand. So either this framework of cognition will be left to the few that are also studying higher-level physics or be left out of the curriculum outright.

I do agree with the article. It shows the uncertainty and complexity that the human mind posses, and how it can be described through such complex means. Although I am skeptical if quantum physics is truly the best framework for understanding the human mind, I think it holds as the biggest contendor at this point in time.

Sleep States

Course:
`../psyc1101-psychology-101/Assignments/sleep_states.pdf`

Document 13

Included from: `../psyc1101-psychology-101/Assignments/sleep_states.pdf`

Illya Starikov

Professor Amanda Burch

General Psychology

June 30, 2025

Sleep States and Memory Processes In Humans: Procedural Versus Declarative Memory Systems

"Sleep on it". This single cliché has permeated society for quite a long time. Although when applied strictly to the context it is usually said (typically when faced with a difficult decision), it does not have merit. Sleep does not help one make a decision no more than a physics textbook helps make you a theoretical physicist; however, just as the book can be a tool to make one a physicist, sleep can be a tool used in the decision making process. We will try to unravel this in this paper.

The relationship for finding correlation between sleep and learning/memory had been, for a long time, something like this:

> "There was substantial evidence from animal studies to support the idea that post-training REM sleep was important for memory consolidation. The human studies, however, provided mixed results. The human studies cited as showing no relationship between sleep and memory utilized simple memorization tasks."

These findings would later come to be proven false. The main reason for the discrepancies came from a naive understanding of how memory worked:

> "These included a very short registration phase taking less than 1 s to occur followed by a period (minutes to hours) after training, during which learned

> material existed as relatively labile short-term memory that was vulnerable to various agents such as strong drugs or electroconvulsive therapy (ECT). As the minutes turned to hours and memory processing continued, the learned material was considered less vulnerable to disruption. After 24 h (or even less), memory formation was considered complete and, thus, no longer vulnerable to disruption. At this point a stable, permanent long-term memory was considered to have been formed. This process has been called consolidation and most workers still use the concept of a permanent "memory trace" existing in the brain"

This poses a problem when testing two different types of learning: "declarative" and "procedural" knowledge. Declarative knowledge "refers to memories accessible to conscious recollection", while procedural "are memories of how to do some skill or how to solve a problem". This is the discrepancy: psychological studies have always assumed that memory was a single, colossal process; however, we know this not to be the case.

There have been four types of studies performed to study the relationship between sleep and memory: "One approach has been to train subjects on some task and then to record the changes in sleep parameters", "deprive the subject of total sleep or of some specific phase or aspect of sleep following task acquisition in order to induce possible memory impairments"„ and finally "A third approach has been to have the subject learn the task either just before bed or during the night at some particular point". These four types have been carried out in regards to declarative material, and the results were, as aforementioned, strongly positive. It had been clearly shown that "after learning a PA [paired associates] task showed superior memory to participants that stayed awake for an equal amount of time”. The research did not stop there — further research showed that total and even selective sleep deprivation was harmful; so much so that you could not tell the selectively sleep deprived and totally sleep deprived apart.

Switching to procedural, patients that underwent study for procedural knowledge were “to engage in cognitive activity that was more complex than simple recall or recognition of previously memorized material. These studies demonstrated a relationship between REM sleep and memory.” Again, the studies were overwhelming positive:

> “Mandai... found an increase in REM sleep duration and number of REM episodes in subjects who underwent 90min training in translation of Morse code. They also reported a high correlation between retention, number of REMs and REM density.”

And this was not limited to just partial sleep deprivation, but also to total and selective sleep deprivation as well. Those who experienced any sleep loss with 48-72 hours also were susceptible to memory loss related to procedural tasks. At this point, there is no denying the relationship between sleep and memory recall; as stated, “The level of recall was positively correlated with the average duration of the NREM-REM sleep cycles”.

This all comes back to the “Sleep on it” statement made previously — although you cannot use sleep a sole deducer for your choices; sleep can make you more effectively a decision maker by allowing you to more quickly recall declarative and procedural knowledge that might be relevant to the question.

I do not think this article is that controversial; humans needs sleep to recharge, reboot, and restore our energy. The article also did not fall into the trap of a “catchall” for sleep, stating that a person needs x-hours of sleep to full function. I also agree with the statements being made; it is quite intuitive that a person will need sleep to recall declarative and procedural knowledge.

Chapter 7

CS 3200 - Numerical Methods

Cribsheet

Course:
`../cs3200-numerical-methods/cribsheet.pdf`

Document 14

Included from: `../cs3200-numerical-methods/cribsheet.pdf`

Numerical Methods Crib Sheet
Illya Starikov

Integration Equations

Note that h usually refers to $(b-a)$. Also, Trapezoidal needs 2 points, Simpson's $^1/_3$ uses 3, Simpson's $^3/_8$ uses 4 and Boole's 5. Also note that for something like $[0,4]$,
$h=4, n=1, h=2, n=2, h=1, n=4$

`Trapezoidal`	$I = \frac{h}{2}\left[f(x_0) + 2\sum_{i=1}^{n} + f(x_n)\right]$
`Richardson`	$I \approx {}^4/_3 I(h_2) - {}^1/_3 I(h_1)$
	$\approx {}^4/_3 I(\text{current cell}) - {}^1/_3 I(\text{previous cell})$
`Romberg`	$I_{j,k} = \frac{4^{k-1} I_{j+1,k-1} - I_{j,k-1}}{4^{k-1}-1}$
	$I_{j,k} = \frac{4^{\text{cell your on - 1}} I_{\text{cell your on}} - I_{\text{cell one up}}}{4^{k-1}-1}$
`Simpson 1/3`	$^h/_6(f(a) + 4f(a+h) + f(b))$
`Simpson 3/8`	$3\,^h/_8(f(a+h) + 3f(a+h) + 3f(a+2h) + f(b))$

Differentiation Equations

Forward Finite-Divided

$$\frac{d}{dx} = \frac{f(x+h) - f(x)}{h} \quad O(h)$$

Backward Finite-Divided

$$\frac{d}{dx} = \frac{f(x) - f(x-h)}{h} \quad O(h)$$

Central Difference

$$\frac{d}{dx} = \frac{f(x+h) - f(x-h)}{2h} \quad O(h^2)$$

$$\frac{d^2}{dx^2} = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \quad O(h^4)$$

Richardson Extrapolation

$$I = I(h) + \mathcal{E}(h)$$

$$I = I(h_1) + \mathcal{E}(h_1) = I(h_2) + \mathcal{E}(h_2)$$

$$\mathcal{E} \approx -\frac{b-a}{2} h^2 \bar{f}''$$

$$\frac{\mathcal{E}(h_1)}{\mathcal{E}(h_2)} \approx -\frac{\frac{b-a}{2} h_1^2 \bar{f}''}{\frac{b-a}{2} h_2^2 \bar{f}''} \approx \frac{h_1^2}{h_2^2}$$

$$\mathcal{E}(h_1) \approx \mathcal{E}(h_2)\left(\frac{h_1}{h_2}\right)^2$$

$$I \approx I(h_1) + \mathcal{E}(h_2)\left(\frac{h_1}{h_2}\right)^2 \approx I(h_2) + \mathcal{E}(h_2)$$

$$\mathcal{E}(h_2) \approx \frac{I(h_2) - I(h_1)}{(^{h_1}/_{h_2})^2 - 1}$$

$$I \approx I(h_2) + \approx \frac{I(h_2) - I(h_1)}{(^{h_1}/_{h_2})^2 - 1}$$

$$I \approx {}^4/_3\, I(h_2) - {}^1/_3\, I(h_1)$$

Differential Equations

Midpoint Method

Note that $y(a) = b \implies x_i = a, y_i = b$.

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_1 + {}^h/_2, y_i + k_1 {}^h/_2)$$

$$y_{i+1} = y_i + k_2 \cdot h$$

Heun Method

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_i + h, y_i + k_1 h)$$

$$y_{i+1} = y_i + \frac{h \cdot (k_1 + k_2)}{2}$$

RK-3

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_i + {}^h/_2, y_i + k_1 {}^h/_2)$$

$$k_3 = f(x_i + h, y_i + (-k_1 + 2k_2)h)$$

$$y_{i+1} = y_i + h \cdot (k_i + 4k_2 + k_3)/6$$

RK-4

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_1 + {}^h/_2, y_i + k_1 {}^h/_2)$$

$$k_3 = f(x_i + {}^h/_2, y_i + k_2 {}^h/_2)$$

$$k_4 = f(x_i + h, y_i + k_3 h)$$

$$y_{i+i} = y_i + h^{(k_1 + 2k_2 + 2k_3 + k_3)}/6$$

Homework 9

Course:
`../cs3200-numerical-methods/homework/homework-9.pdf`

Document 15

Included from: `../cs3200-numerical-methods/homework/homework-9.pdf`

Homework #9

Illya Starikov

Due Date: December 1st, 2016

1 Simpson $^1/_3$ Rule For $1 + 2x + 3x^2$

$$\begin{aligned}
\int_0^4 1 + 2x + 3x^2 \, dx &\approx {}^h/_6(f(a) + 4\, f(a+h) + f(b)) \\
&= {}^4/_6(1 + 4(1 + 4 + 12) + (1 + 8 + 48)) \\
&= {}^2/_3(1 + 68 + 57) \\
&= 84
\end{aligned}$$

2 Simpson's $^3/_8$ Rule For x^3

$$\begin{aligned}
\int_0^3 x^3 \, dx &\approx {}^3/_8 h(f(a) + 3(a+h) + 3(a+2h) + f(b)) \\
&= {}^3/_8(0 + 1 + 8 + 27) \\
&= {}^3/_8 \cdot 36 \\
&= 13.5
\end{aligned}$$

3 Trapezoidal, Richardson and Romberg Method of x^4

We know the equation for the trapezoid rule to be

$$I = \frac{h}{2}\left[f(x_0) + 2\sum_{i=1}^{n} + f(x_n)\right]$$

From this, we obtain

$$\begin{aligned}
I_4 &= \frac{4}{2}\left(f(0) + f(4)\right) = 512 \\
I_2 &= \frac{2}{2}\left(f(0) + 2\,f(2) + f(4)\right) = 288 \\
I_1 &= \frac{1}{2}\left(f(0) + 2\,f(2) + 2\,f(3) + f(4)\right) = 226
\end{aligned}$$

For Richardson, we calculate the terms as follows:

$$\begin{aligned}
I &\approx {}^4\!/_3 I(h_2) - {}^1\!/_3 I(h_1) \\
I_2 &= {}^4\!/_3(288) - {}^1\!/_3(512) = {}^{640}\!/_3 = 213.\bar{3} \\
I_1 &= {}^4\!/_3(226) - {}^1\!/_3(288) = {}^{616}\!/_3 = 205.\bar{3}
\end{aligned}$$

And for Romberg,

$$\begin{aligned}
I_{j,k} &= \frac{4^{k-1} I_{j+1,k-1} - I_{j,k-1}}{4^{k-1}} \\
I_1 &= \frac{4^2({}^{616}\!/_3) - {}^{640}\!/_3}{15} = \frac{1024}{5} = 204.8
\end{aligned}$$

We observe that the table looks like

h	**trapazoidal**	**Richardson**	**Romberg**
4	512		
2	288	$213.\bar{3}$	
1	226	$205.\bar{3}$	204.8

4 Derivation of Richardson Extrapolation

We recall that the value of integration I is equal to approximation + error, which we can eloquently write

$$I = I(h) + \mathcal{E}(h) \tag{1}$$

Where $\mathcal{E}$ represents our error and $I(h)$ represents our approximation. Supposing we have two different steps size (h_1 and h_2), we can rewrite equation (1) in the form

$$I = I(h_1) + \mathcal{E}(h_1) = I(h_2) + \mathcal{E}(h_2) \tag{2}$$

Now, let us expand $\mathcal{E}(h)$ from equation (1) and equation (2).

$$\mathcal{E} \approx -\frac{b-a}{2} h^2 \, \bar{f}'' \tag{3}$$

Where b and a are the upper and lower bounds, respectively, and $\bar{f}''$ is the average value of $\frac{d}{dx} f(x)$ (where $f(x)$ is the function we are integrating). Therefore, we can a ratio of the two errors

$$\frac{\mathcal{E}(h_1)}{\mathcal{E}(h_2)} \approx -\frac{\frac{b-a}{2} h_1^2 \, \bar{f}''}{\frac{b-a}{2} h_2^2 \, \bar{f}''} \approx \frac{h_1^2}{h_2^2}$$

Which can be algebraically manipulated to

$$\mathcal{E}(h_1) \approx \mathcal{E}(h_2) \left(\frac{h_1}{h_2}\right)^2 \tag{4}$$

Now we can substitute equation (4) back into equation (2) to obtain

$$I \approx I(h_1) + \mathcal{E}(h_2) \left(\frac{h_1}{h_2}\right)^2 \approx I(h_2) + \mathcal{E}(h_2)$$

Which can be solve for

$$\mathcal{E}(h_2) \approx \frac{I(h_2) - I(h_1)}{(h_1/h_2)^2 - 1}$$

Now that we have an estimate (hence the $\approx$) of the truncation error, We once plug this back into equation (2) to get

$$I \approx I(h_2) + \approx \frac{I(h_2) - I(h_1)}{(h_1/h_2)^2 - 1}.$$

Because we know the interval to be halved, we can safely assume $h_2 = h_1/2$. Thus, our equation reduces to

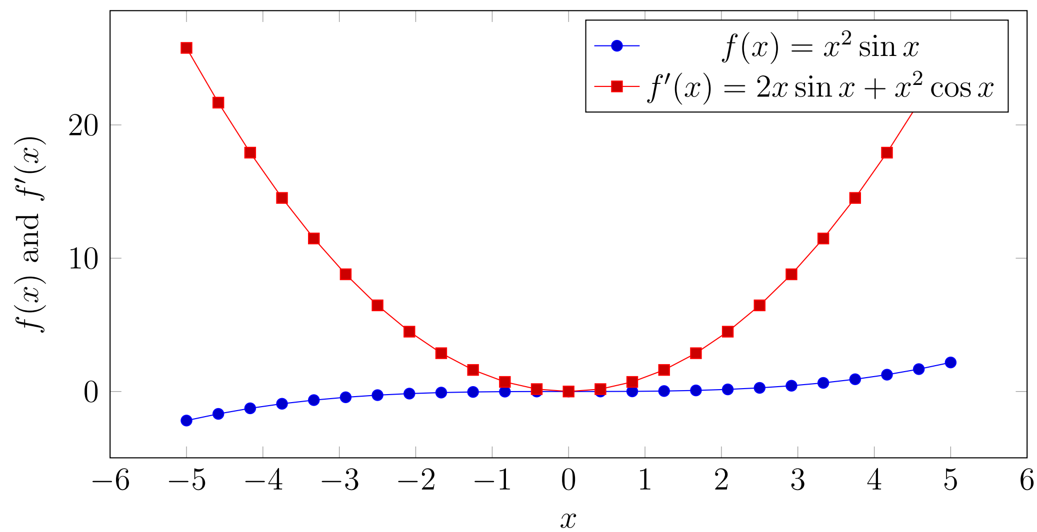
$$I \approx {}^{4}\!/_{3}\, I(h_2) - {}^{1}\!/_{3}\, I(h_1) \tag{5}$$

5 Derivative of $x^2 \sin x$

From the chain rule, we get

$$\frac{d}{dx} x^2 \sin x = 2x \sin x + x^2 \cos x$$

It can be plotted as follows



To compute the two derivatives for $h_1 = 0.2$ and $h_2 = 0.1$, we use the centered difference formulas. For h_1

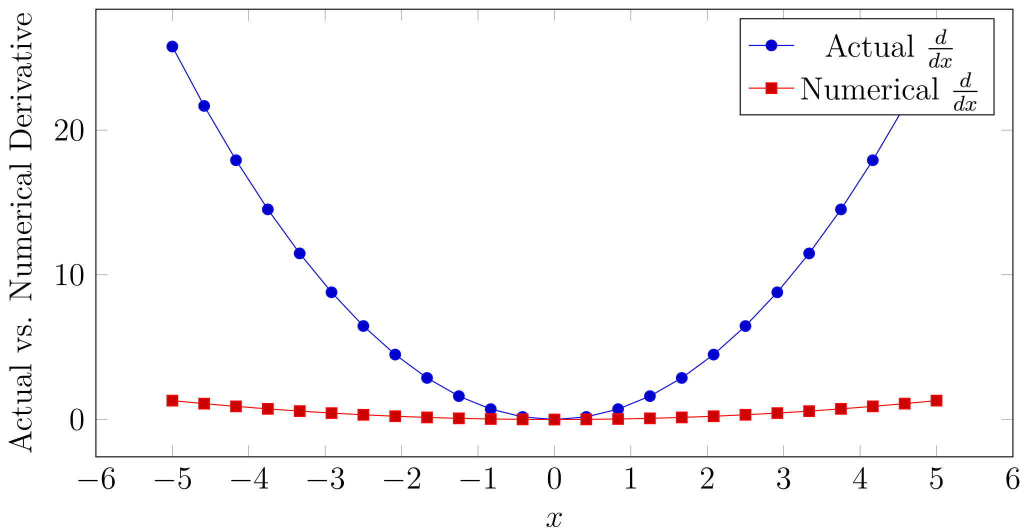
$$\begin{aligned}
\frac{d}{dx} &= \frac{f(x_{i+1}) - f(x_{i-1})}{2h} \\
&= \frac{f(x+h) - f(x-h)}{2h} \\
&= \frac{f(x + {}^{1}\!/_{5}) - f(x - {}^{1}\!/_{5})}{{}^{2}\!/_{5}} \\
&= \frac{(x + {}^{1}\!/_{5})^2 \sin(x + {}^{1}\!/_{5}) - ((x - {}^{1}\!/_{5})^2 \sin(x - {}^{1}\!/_{5}))}{{}^{2}\!/_{5}}
\end{aligned}$$

Similarly, for h_2

$$\frac{(x+{}^1\!/_{10})^2\sin(x+{}^1\!/_{10})-((x-{}^1\!/_{10})^2\sin(x-{}^1\!/_{10}))}{{}^1\!/_5}$$

We combine these for obtain the Richardson extrapolation formula,

$$\begin{aligned}
D &\approx {}^4\!/_3 D(h_2) - {}^1\!/_3 D(h_1) && (6)\\
&\approx {}^4\!/_3 \frac{(x+{}^1\!/_{10})^2\sin(x+{}^1\!/_{10})-((x-{}^1\!/_{10})^2\sin(x-{}^1\!/_{10}))}{{}^1\!/_5} && (7)\\
&- {}^1\!/_3 \frac{(x+{}^1\!/_5)^2\sin(x+{}^1\!/_5)-((x-{}^1\!/_5)^2\sin(x-{}^1\!/_5))}{{}^2\!/_5} && (8)
\end{aligned}$$



6 Central Difference of x^4

6.1 First Order

From the power rule, we notice that

$$\frac{d}{dx}x^4 = 4x^3 \implies f'(1) = 4$$

Using centered difference formula,

$$
\begin{aligned}
\frac{d}{dx} &= \frac{f(x_{i+1}) - f(x_{i-1})}{2h} \\
&= \frac{(1 + {}^{1}\!/\!_{2})^4 - (1 - {}^{1}\!/\!_{2})^4)}{(2{}^{1}\!/\!_{2})} \\
&= 5
\end{aligned}
$$

We notice the error to be $|\frac{4-5}{4}| = |-.25| \implies \mathbf{25}\%$

6.2 Second Order

From the power rule, we notice that

$$\frac{d^2}{dx^2}x^4 = 12x^2 \implies f'(1) = 12.$$

Using centered difference formula,

$$
\begin{aligned}
\frac{d^2}{dx^2} &= \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{h^2} \\
&= \frac{(1 + {}^{1}\!/\!_{2})^4 - 2(1^4) + (1 - {}^{1}\!/\!_{2})^4)}{({}^{1}\!/\!_{2})^2} \\
&= {}^{25}\!/\!_{2}
\end{aligned}
$$

We notice the error to be $|\frac{12-25/2}{12}| = |-.041\bar{6}| \implies \mathbf{4.16}\%$.

Homework 10

Course:
`../cs3200-numerical-methods/homework/hw-10.pdf`

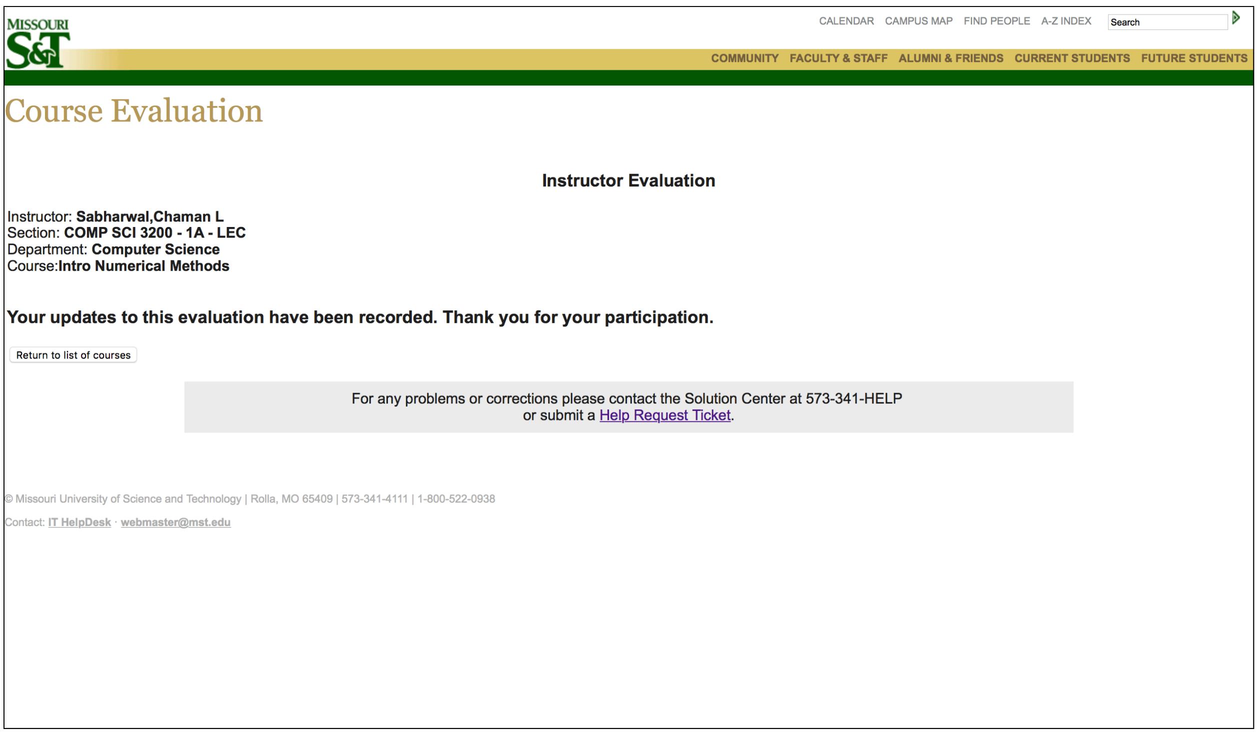
Document 16

Included from: `../cs3200-numerical-methods/homework/hw-10.pdf`

Homework #10

Illya Starikov

Due Date: December 6^{th}, 2016



Lecture 1

Course: `../cs3200-numerical-methods/lecture1.pdf`

Document 17

Included from: `../cs3200-numerical-methods/lecture1.pdf`

1 Introduction

- The differential equation for newton's equation is $\frac{dv}{dt} = g - \frac{cv}{m}$, where cv is the drag coefficient $\times$ velocity.

$$\frac{dv}{dt} = g - \frac{cv}{m} \tag{1}$$

$$dv = (g - \frac{cv}{m})dt \tag{2}$$

$$\int \frac{1}{g - \frac{cv}{m}} dv = \int dt \tag{3}$$

$$\lg(g - \frac{cv}{m}) = t + c \tag{4}$$

$$v = mg/c(1 - e^{c/mt}) \tag{5}$$

Lecture 2

Course: `../cs3200-numerical-methods/lecture2.pdf`

Document 18

Included from: `../cs3200-numerical-methods/lecture2.pdf`

2 Error Tolerance

- There will be 15% extra credit.
- Two important terms, accuracy & precision.
  - Accuracy refers to how close the computed value is to the true value.
  - Precision means how close the values are together.
- Because we want everything relative, we calculate with two different errors.
  - Absolute error = $e_a = |\text{True value} - \text{Approximation}|$
  - Relative error = $\epsilon_t = \frac{\text{True value}-\text{Approximation}}{\text{true}} \times 100$
- These equations can be used for precision as well.
  - $e_a = |\text{previous} - \text{current}|$
  - $\epsilon_t = \frac{\text{previous}-\text{current}}{\text{current}} \times 100$
- Stopping point is when it's below a certain threshold.
- If true value is 0, shift over by 1.
- For a result to be accurate or precise, should be $< 0.5 \times 10^{2-n}$, where n is the number of significant digits.
- the function of e^x can be represented as $e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \cdots + \frac{x^n}{n!}$
  - for $x = 1$, $e^1 = 2.71828$
  - for $x = -1$, $a = 1 - 1 = 0$

Lecture 4

Course: `../cs3200-numerical-methods/lecture4.pdf`

Document 19

Included from: `../cs3200-numerical-methods/lecture4.pdf`

4 Roots of Equations

- The bisection method is one way to find the root.
- There are two requirements.
  1. $f(x)$ is continuous on $[a, b]$
  2. $f(a) \times f(b) = -(\text{something})$

Numerical Methods Equations

Course:
`../cs3200-numerical-methods/numerical-methods-equations.`

Document 20

Included from:
`../cs3200-numerical-methods/numerical-methods-equations.pdf`

Numerical Methods Equations

Illya Starikov

June 30, 2025

1 Taylor Series

$$
\begin{aligned}
f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^{(3)}(a)}{3!}(x-a)^3 + \cdots \quad (1)\\
= \sum_{k=0}^{\infty} \frac{f^{(k)}(a)}{k!}(x-a)^k. \quad (2)
\end{aligned}
$$

2 Taylor Series of e^x

$$e^x = 1 + x + \frac{x^2}{2!} + \cdots \quad (3)$$

$$\frac{1}{e^{-x}} = \frac{1}{1 - x + \frac{x^2}{2} + \cdots} \quad (4)$$

3 False Position

$$r = b - \frac{(b-a)\cdot f(b)}{f(b)-f(a)} \quad (5)$$

4 Newton's Method

$$x_{n+1} = x_n + \frac{f(x_n)}{f'(x_n)} \quad (6)$$

5 Secant Lines

$$x_3 = x_1 - \frac{(x_2 - x_1)\cdot f(x_1)}{f(x_2) - f(x_1)} \quad (7)$$

6 Modified Secant Method

$$x_2 = x_1 - \frac{f(x_1)\Delta}{f(x_1 + \Delta) - f(x_1)} \tag{8}$$

Chapter 8

CS 4096 - Software Systems Development

Ethics

Course:
`../cs4096-software-systems-development/ethics.pdf`

Document 21

Included from: `../cs4096-software-systems-development/ethics.pdf`

Ethical, Security, Legal, and Social Impact

CS 4096, Dr. Morales

Team Splatoonio*

June 30, 2025

War Paint is an augmented reality application that utilizes the player's current geographic location as a stylus to compete against other users. As part of gameplay this application collects user data and displays some for other users to see. While applications that give others your current location exist (i.e., SnapChat or Find My Friends) none of these display your location to anonymous users. In order to play War Paint the user must accept that their current location is being shared with other users; this creates an ethical conundrum. First, it would be illegal to allow children 13 years or younger to use the application without parent permission. There is a legal gray area in sharing a user's location from age 13 to 18, so to mitigate these ethical concerns, the application will likely be restricted to users 18 years or older.

The creators of War Paint also have an ethical responsibility to preserve the security of user data. War Paint will likely collect user data for the purpose of targeted advertisement; however, this data will never be sent to the War Paint server. In other words, we do not permanently store user data for longer than is necessary to play the game, and the data we do collect is all anonymous. It is important that this, as well as user passwords and profile information, remain secure and confidential. Passwords are currently stored on the server in plaintext due to time constraints, but before release, the server would be switched over to using a standard salted hashing function to protect user data. Clients would also be forced to send communication over HTTPS ensuring their connections were not visible in transit. In addition, most, if not all, of our server endpoints are also vulnerable to SQL injection, and would be fixed before releasing to mobile app stores. We would also like to have someone perform a penetration test of our system to help verify its security.

New users would be required to read and accept an End-User License Agreement stating that they are at least 18 years of age. It will clarify that the War Paint mobile app will collect user data and display the user's current location anonymously to other

*Adam Evans, Illya Starikov, Ian Howell, Deacon Seals, Michael Harrington, Luke Parton, Eric Michalak

players while the game is in session. It will state that the creators and owners of War Paint are not responsible for damages or risks incurred by sharing the user's location.

The EULA will also clarify that users are responsible for their own physical safety while playing. War Paint is a physically active game. The most successful way to play is to maintain the highest possible speed for the duration of a game session. This would encourage especially competitive users to take risks in order to succeed. To prevent people from using cars or other motorized vehicles, a strict speed limit of 20 miles per hour will be imposed on the players. If this speed limit is exceeded, the user will cease painting the map until their speed falls back to acceptable levels. Repeatedly violating the speed limit will result in a temporary ban. In addition, there will be a report system that allows players to indicate which players are participating in dangerous behaviors while playing.

After witnessing the social impact Pokémon GO had on cities and parks, it is hard to estimate the potential impact of War Paint. On a smaller scale it could increase the number of people running on college campuses. Encouraging college students to exercise with their peers would be a positive social impact. On a larger scale this game could cause congestion in areas where it is the most popular. Matchmaking could keep track of the number of users in any specific area to prevent too many people from playing in too small a location.

This page is intentionally left blank.

Reflection Essay

Course:
../cs4096-software-systems-development/reflection-essay.

Document 22

Included from:
../cs4096-software-systems-development/reflection-essay.pdf

Reflection Essay

Senior Design (Comp Sci 4096)

Illya Starikov

Due Date: Wednesday, December 13$^{\text{th}}$

Throughout the course of Senior Design, I have learned a lot of useful information that will translate not only to the remainder of school, but to the actual development in the real world. Along the way of creating War Paint, I have learned several lessons that I hope to take with me upon graduating college.

For backstory, War Paint is a real-time, augmented reality game who's primary objective is to traverse as much of a terrain as possible. Specifically,

1. The player is boxed in a particular perimeter. Traversing outside said parameter is strictly forbidden (and will not count towards team score).
2. Players are split into even teams. As a player moves around the boxed perimeter, "paint" is placed down (colored with the team's primary color, red or blue).
3. Paining over another team's paint voids the other team's paint, and contributes only to who's team has the last layer of paint on the field.
4. At the end of a designated period of time, the team with the most paint on the board wins.

During the semester, I got to work with a team that had to deal with a whole stack. Our particular team had

- An Android Team
- An iOS Team
- A R&D Team (to focus on potential of hardware)
- A Server Team

Because my specialty was iOS Development, I took the role of working on the iOS application. Most of the software development process was simple: design the Model View Controllers (MVCs). The MVCs already had a hierarchy to them, so barely any architecture work had to be done.

The takeaway I gained from this class wasn't a particular tool or methodology; it was a particular mindset. Most of my group projects up to this point have placed me in a leadership role of some sorts. Because of my busy schedule this semester, I was not comfortable in that position. Because of this, I had to learn how to coordinate with others, be a team player, and put my faith into a different team lead.

My knowledge of the Software Development Process has not particularly changed up to this point. I have held several internships and have worked on massive projects before; most of my knowledge had been learned outside of the classroom.

The most important lesson I learned was how to not bite off more than I can chew. I had to usually take the minimal amount of work per week, for I had priorities in other classes and my job. This will greatly impact the way I handle work, where a work-life balance will be hard to maintain. I hope to keep this lesson in the back of my mind for future reference.

Chapter 9

CS 5400 - Artificial Intelligence

AI Master

Course:
`../cs5400-artificial-intelligence/master.pdf`

Document 23

Included from: `../cs5400-artificial-intelligence/master.pdf`

Introduction To Artificial Intelligence

Illya Starikov

Last Modified: June 30, 2025

1 Introduction To AI

- What is AI?
  - Systems that act like humans
  - Systems that think like humans
  - Systems that think rationally
  - Systems that act rationally

2 Intelligent Agents

- Computer agents...
  - Perceive environment
  - Operate autonomously
  - Persist over prolonged periods
- Rational Agents...
  - Are affected by their environment.
  - Use sensors to interpret their environment
  - From said sensors, it acts on them via actuators
  - These are summarized by Figure 1. The main focus of the semester is filling in the `?`.

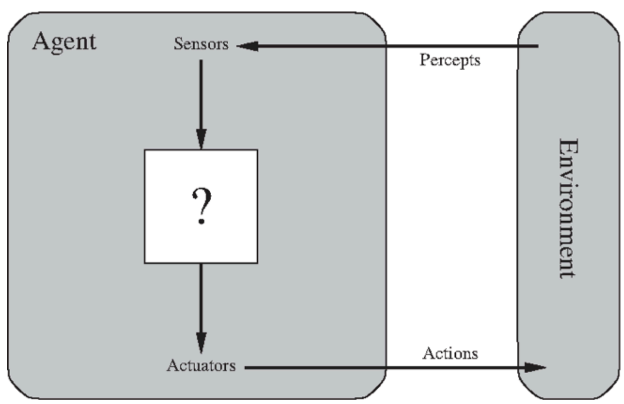


Figure 1: An Agent Cycle

- Rational behavior depends on...
  - Agent's performance measure
  - Agent's prior knowledge
  - Possible percepts and actions
  - Agent's percept sequence
- We define a rational agents as follows:

> "For each possible percept sequence, a rational agent selects an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and any prior knowledge the agent has."

- PEAS[1] descriptoin and properties…

  Observable/Partially Observable If it is possible to determine the complete state of the environment at each time point from the percepts it is observable; otherwise it is only partially observable.

  Deterministic/Stochastic/Strategic If the next state of the environment is completely determined by the current state and the action executed by the agent, then it is determinstic. If the environment is deterministic except for the actions of other agents, then the environment is strategic. If it is randomly determined, then it is stochastic.

  Episodic/Sequential In an episodic environment, each episode consists of the agent perceiving and then acting. The quality of its action depends just on the episode itself. Subsequent episodes do not depend on the actions in the previous episodes. Episodic environments are much simpler because the agent does not need to think ahead.

  Static/Semi-Dynamic/Dynamic If an environement is unchanged while an agent is deliberating, it is static. The environment is semi-dynamic if the environment itself does not change with the passage of time but the agent's performance score does. If the environment is constantly changing, then it is dynamic.

  Discrete/Continuous If there are a limited number of distinct, clearly defined, states of the environment, the environment is discrete (For example, chess); otherwise it is continuous (For example, driving).

  Discrete/Continuous If there are a limited number of distinct, clearly defined, states of the environment, the environment is discrete (For example, chess); otherwise it is continuous (For example, driving).

  Single Agent/Multi-Agent The environment may contain other agents which may be of the same or different kind as that of the agent.

  Competitive/Cooperative If agents are in direct competition with each other (i.e., chess), they are competitive. If they have a common goal, they are cooperative.

  Known/Unknown An environment is considered to be known if the agent understands the laws that govern the environment's behavior. For example, in chess, the agent would know that when a piece is "taken" it is removed from the game.

- Some agent types include…

  Simple Reflex Agents Simple reflex agents act only on the basis of the current percept, ignoring the rest of the percept history. The agent function is based on the condition-action rule: `if condition then` action.

[1] Performance Measure, Environment, Actuators, and Sensors

Model-Based Reflex Agents A model-based agent can handle partially observable environments. Its current state is stored inside the agent maintaining some kind of structure which describes the part of the world which cannot be seen. This knowledge about "how the world works" is called a model of the world, hence the name "model-based agent".

Goal-Based Agents Goal-based agents further expand on the capabilities of the model-based agents, by using "goal" information. Goal information describes situations that are desirable. This allows the agent a way to choose among multiple possibilities, selecting the one which reaches a goal state. Search and planning are the subfields of artificial intelligence devoted to finding action sequences that achieve the agent's goals.

Utility-Based Agents Goal-based agents only distinguish between goal states and non-goal states. It is possible to define a measure of how desirable a particular state is. This measure can be obtained through the use of a utility function which maps a state to a measure of the utility of the state. A more general performance measure should allow a comparison of different world states according to exactly how happy they would make the agent. The term utility can be used to describe how "happy" the agent is.

Learning Agents Learning has the advantage that it allows the agents to initially operate in unknown environments and to become more competent than its initial knowledge alone might allow. The most important distinction is between the "learning element", which is responsible for making improvements, and the "performance element", which is responsible for selecting external actions.

- The definition of a problem solving agent is as follows…

> Problem-solving agents are goal based agents that decide what to do based on an action sequence leading to a goal state.

3 Introduction To Search

- Open-loop problem solving steps are as follows...
  1. Problem-formulation (actions & states)
  2. Goal-formulation (states)
  3. Search (action sequences)
  4. Execute solution
- All well-defined problems have the following...
  - An initial state
  - An actions set (usually denoted by ACTIONS(s))
  - A transition model (usually denoted by RESULT(s, a))
  - A goal test
  - A step cost (usually denoted by C(s, a, s'))
  - Path cost
  - Solution / optimal solution[2]
- Search trees...
  - Root corresponds to initial state
  - Search algorithms iterate through goal testing and expanding a state until goal is found
  - Order of state expansion is critical

```
function TREE-SEARCH(problem)
    frontier ← using initial problem state

    loop
        if EMPTY(frontier) then
            return Fail
        end if

        choose leaf node and remove it from frontier

        if chosen node contains goal state then
            return corresponding solution
        end if
```

[2]The optimal solution is just the solution with the least path cost

```
        expand chosen node and add resulting nodes to frontier
    end loop
end function
```

- The keyword **choose** is critical; there are different ways to choose nodes, providing better search algorithms.
- However, this doesn't deal with repeated students, redundant paths, or loops. For graphs, we have as follows.

```
function GRAPH-SEARCH(problem)
    frontier ← using initial problem state
    explored set ← {}
    loop
        if EMPTY(frontier) then
            return Fail
        end if
        choose leaf node and remove it from frontier
        if chosen node contains goal state then
            return corresponding solution
        end if
        explored set ← explored set ∪ chosen node
        if chosen node ∉ frontier or explored set then
            expand chosen node and add resulting nodes to frontier
        end if
    end loop
end function
```

- Search node data structure
  - n.State
  - n.Parent-Node
  - n.Action
  - n.Path-Cost
- ***States are not search nodes***
- We define a child nodes as follows.

```
function CHILD-NODE(problem, parent, action)
```

```
    return node with:
        STATE = problem.RESULT(parent.STATE, action)
        PARENT = parents
        ACTION = action
        PATH-COST = parents.PATH-COST + problem.STEP-COST(parent.STATE, action)
end function
```

- The frontier…
  - Is a set of leaf nodes
  - Implemented as a queue with operations…
    - EMPTY
    - POP
    - INSERT
  - Queue types: FIFO, LIFE (stack), and priority queue.
- Explored set…
  - Set of expanded nodes
  - Implemented typically as a hash table for constant time insertion and lookup
- Problem-solving performance…
  - Completeness
  - Optimality
  - Time complexity
  - Space complexity
- Complexity in AI…

b Branching Factor

d Depth of Shallowest Goal Node

m Max Path Length in State Space

Time Complexity Generated Nodes

Space Complexity Max Number of Nodes Stored

Search Cost Time + Space Complexity

Total cost Search + Path Cost

4 Uninformed Search Algorithms

An uninformed (a.k.a. blind, brute-force) search algorithm generates the search tree without using any domain specific knowledge.

4.1 Breadth First Tree Search (BFTS)

- Frontier is a FIFO Queue
- Complete if b and d are finite
- Optimal if path-cost is non-decreasing function of depth
- Time complexity is $\mathcal{O}\left(b^d\right)$
- Space complexity is $\mathcal{O}\left(b^d\right)$

```
function Breadth-First-Search(problem)
    node ← a node with State = problem.Initial-State,Path-Cost = 0
    if problem.Goal-Test(node.State) then
        return Solution(node)
    end if
    frontier ← a FIFO queue with node as the only element
    explored ← an empty set
    loop
        if Empty?(frontier) then
            return failure
        end if
        node ← Pop(frontier)                    ▷ Chooses the shallowest node in frontier
        explored ← explored ∪ node.State
        for all action ∈ problem.Actions(node.State) do
            child ← Child-Node(problem, node, action)
            if child.State ∉ explored, frontier then
                if problem.Goal-Test(child.State) then
                    return Solution(child)
                end if
                frontier ← Insert(child, frontier)
```

```
            end if
        end for
    end loop
end function
```

4.2 Uniform Cost Search

- $g(n)$ is the lowest path-cost from start node to node n
- Frontier is a priority queue ordered by $g(n)$

```
function UNIFORM-COST-SEARCH(problem)
    node ← a node with STATE = problem.INITIAL-STATE,PATH-COST = 0
    frontier ← a priority ordered by PATH-COST, with node as the only element
    explored ← an empty set

    loop
        if EMPTY?(frontier) then
            return failure
        end if

        node ← POP(frontier)                    ▷ Chooses the shallowest node in frontier
        explored ← explored ∪ node.STATE

        for all action ∈ problem.ACTIONS(node.STATE) do
            child ← CHILD-NODE(problem, node, action)

            if child.STATE ∉ explored, frontier then
                if problem.GOAL-TEST(child.STATE) then
                    return SOLUTION(child)
                end if

                frontier ← INSERT(child, frontier)
            end if
        end for
    end loop
end function
```

4.3 Depth First Tree Search (DFTS)

- Frontier: LIFO queue (Stack)
- Not Complete

- Not Optimal
- Time Complexity: $\mathcal{O}(b^m)$
- Space complexity: $\mathcal{O}(bm)$
- There exists a backtracking version
  - Space Complexity: $\mathcal{O}(b^m)$
  - Modifies rather than copies state description.
- The implementation is identical to Breath First Search, except the Frontier.

4.4 Depth-Limited Tree Search (DLTS)

- Frontier: LIFO queue (Stack)
- Not Complete When $l < d$
- Not Optimal
- Time Complexity: $\mathcal{O}(bl)$
- Space complexity: $\mathcal{O}(bl)$
- Diameter: Min number of steps to get from any state to any other state.

```
function Depth-Limited-Search(problem, limit)
    return Recursive-DLS(Node(problem.Initial-State), problem, limit)
end function
function Recursive-DLS(node, problem, limit)
    if problem.Goal-Test(node.State) then
        return Solution(node)
    else if limit = 0 then
        return cutoff
    else
        cutoff occurred? ← false
        for Action ∈ problem.Actions(node.State) do
            child ← Child-Node(problem, node, action)
            result ← Recursive-DLS(child, problem, limit - 1)
            if result = cutoff then
                cutoff occurred?true
            else if result ≠ failure then
```

```
                return result
            end if
        end for

        if cutoff occurred? then
            return cutoff
        else
            return failure
        end if
    end if
end function
```

4.5 Iterative-Deepening Depth-First Tree Search (ID-DFTS)

- Complete if b is finite
- Optimal if path-cost is non-decreasing function of depth
- Time Complexity: $\mathcal{O}\left(b^d\right)$
- Space Complexity: $\mathcal{O}\left(()\, bd\right)$

```
function ITERATIVE-DEEPENING-SEARCH(problem)
    for depth = 0 to ∞ do
        result ← DEPTH-LIMITED-SEARCH(problem, depth)

        if result ≠ cutoff then
            return result
        end if
    end for
end function
```

4.6 Bidirectional Breath First Search (BiBFTS)

- Complete if b is finite
- Not optimal "out of the box"
- Time Complexity: $\mathcal{O}\left(b^{d/2}\right)$
- Space Complexity: $\mathcal{O}\left(b^{d/2}\right)$

5 Informed Search Algorithms

An informed search algorithm generates the search tree with using any domain specific knowledge.

Note, for these algorithms,

$$\begin{aligned} g(n) &= \text{lowest path-cost from start node to node } n \\ h(n) &= \text{estimated non-negative path-cost of cheapest path from node } n \text{ to a goal node} \\ &= \text{with } h(goal) \end{aligned}$$

where,

- $h(n)$ is a heuristic function.
- Heuristics incorporate problem-specific knowledge.
- Heuristics need to be relatively efficient to compute.
- The function $h(n)$ is admissible if

  $$\forall n \, (h(n) \leq C^*(n))$$

  In other words, a heuristic is admissible if the estimated cost is never more than the actual cost from the current node to the goal node.

- The function $h(n)$ is consistent if

  $$\forall n, n' \, (h(n) \leq c(n, a, n') + h(n'))$$

  In other words, a heuristic is consistent if the cost from the current node to a successor node, plus the estimated cost from the successor node to the goal is less than or equal to the estimated cost from the current node to the goal.

For the major search search algorithms:

$$\begin{aligned} \text{UCS} : f(n) &= g(n) \\ \text{GBeFS} : f(n) &= h(n) \\ \text{A*S} : f(n) &= g(n) + h(n) \end{aligned}$$

5.1 A^* Proof

Theorem 1. *A^* TS employing heuristic $h(n)$ is optimal if $h(n)$ is admissible.*

Proof. Suppose suboptimal goal node G appears on the frontier and let the cost of the optimal solution be C^*. From the definition of $h(n)$ we know that $h(G) = 0$ and because G is suboptimal we know $f(G) > C^*$. Together this gives $f(G) = g(G) + \cancel{h(G)} = g(G) > C^*$.

If there is an optimal solution, then there is a frontier node N that is on the optimal solution path. Our proof is for an admissible heuristic, so we know $h(N)$ does not overestimate; thus $f(N) = g(N) + h(N) \leq C^*$.

Together this given $f(N) \leq C^* < f(G) \implies f(N) < f(G)$.

As A^*TS expands lower f-cost nodes before higher f-cost nodes, N will always be expanded before G. Thus, A^*TS is optimal. $\square$

5.2 Consistent $\implies$ Admissible

Theorem 2. *Every consistent heuristic is admissible*

Proof. Take an arbitrary path to an arbitrary goal node G denoted $S_1, S_2, S_3, \ldots, S_G$. As h is consistent, we know $\forall n, n' : h(n) \leq c(n, a, n') + h(n')$.

$$
\begin{array}{l|l}
S_1\, S_2 & h(S_1) - h(S_2) \leq c(S_1, S_2) \\
S_2\, S_3 & h(S_2) - h(S_3) \leq c(S_2, S_3) \\
\vdots & \vdots \\
S_{G-1}\, S_G & h(S_{G-1}) - h(S_G) \leq c(S_{G-1}, S_G) \\
\hline
S_1\, S_G & h(S_1) \leq c(S_1, S_G) \\
S_2\, S_G & h(S_2) \leq c(S_2, S_G) \\
\vdots & \vdots \\
S_k\, S_G & h(S_k) \leq c(S_k, S_G)
\end{array}
$$

$\square$

As this is true for all goal nodes S_G, it's also true for the optimal goal node. Thus,

$$\forall h \in \{1, 2, \ldots, h\}, h(S_k) \leq C^*(S_k)$$

which is the definition of admissibility.

5.3 Best First Search

- Select node to expand based on evaluation function $f(n)$.
- Node with lowest $f(n)$ selected as $f(n)$ correlated with path-cost.
- Represent frontier with priority queue sorted in ascending order of f-values.

5.4 Greedy Best First Search

- Incomplete (so also not optimal).
- Worst-case time complexity and space complexity: $\mathcal{O}(b^m)$.
- Actual complexity depends on accuracy of $h(n)$.

5.5 A^* Search

- $f(n) = g(n) + h(n)$.
- $f(n)$ is estimated cost of optimal solution through node n
- If $h(n)$ satisfies certain conditions, $A*$ Search is complete and optimal
- Is Optimally efficient for consistent heuristics.
- Run-time is a function of the heuristic error.
- A^* Graph Search not scalable due to memory requirements.

6 Adversarial Search

An adversarial environment is characterized by:

- Competitive multi-agent
- Turn-taking

The simplest types are:

- Discrete
- Deterministic
- Two-Player
- Zero-Sum Games
- Perfect Information

Now, the search problem is described by:

S_0 Initial State

$\textsc{Player}(\textsc{s})$ Which player has the move?

$\textsc{Actions}(\textsc{s})$ Set of legal moves.

$\textsc{Result}(\textsc{s, a})$ Defines transitional model.

$\textsc{Terminal-Test}(\textsc{s})$ Is this a game over state?

$\textsc{Utility}(\textsc{s})$ Associates player-dependent values with terminal states.

6.1 Minimax

- Time complexity: $\mathcal{O}(b^m)$.
- Space complexity: $\mathcal{O}(b\,m)$.

$$\textsc{Minimax}(\textsc{s}) = \begin{cases} \textsc{Max'sUtility}(\textsc{s}) & \leftrightarrow \textsc{Terminal-Test}(\textsc{s}) \\ \max_{a \in \textsc{Actions}(\textsc{s})} \textsc{Minimax}(\textsc{Result}(\textsc{s, a})) & \leftrightarrow \textsc{Player}(\textsc{s}) = \textsc{Max} \\ \min_{a \in \textsc{Actions}(\textsc{s})} \textsc{Minimax}(\textsc{Result}(\textsc{s, a})) & \leftrightarrow \textsc{Player}(\textsc{s}) = \textsc{Min} \end{cases}$$

Just as there is Depth-Limited Depth First Search, there's Depth-Limited Minimax:

- State Evaluation Heuristic estimates Minimax value of a node.
- Note that the Minimax value of a node is always calculated for the Max player, even when the Min player is at move in that node.

$$\textsc{Minimax}(\textsc{s},\ \textsc{d}) = \begin{cases} \textsc{Eval}(\textsc{s}) & \leftrightarrow \textsc{Cuttoff-Test}(\textsc{s}) \\ \max_{a\in\textsc{Actions}(\textsc{s})} \text{H-}\textsc{Minimax}(\textsc{Result}(\textsc{s},\ \textsc{a}),\ \textsc{d}+1) & \leftrightarrow \textsc{Player}(\textsc{s}) = \textsc{Max} \\ \min_{a\in\textsc{Actions}(\textsc{s})} \text{H-}\textsc{Minimax}(\textsc{Result}(\textsc{s},\ \textsc{a}),\ \textsc{d}+1) & \leftrightarrow \textsc{Player}(\textsc{s}) = \textsc{Min} \end{cases}$$

6.2 State Evaluation

A good state evaluation heuristic should:

1. Order the terminal states in the same way as the utility function.
2. Be relatively quick to compute.
3. Strongly correlate non-terminal states with chance of winning.

Also, to take the weighted linear state evaluation heuristic:

$$\textsc{Eval}(\textsc{s}) = \sum_{i=1}^{n} w_i\, f_i(s) \tag{1}$$

6.3 Alpha-Beta Pruning

Some terminology:

α Worst value that Max will accept at this point of the search tree.

β Worst value that Min will accept at this point of the search tree.

Fail-Low Encountered value $\leq \alpha$.

Fail-High Encountered value $\geq \beta$.

Prune If fail-low for Min-player or fail-high for Max-player.

With Alpha-Beta Pruning, we get the following complexities:

- Worst-Case: $\mathcal{O}\left(b^d\right)$.
- Best-Case: $\mathcal{O}\left(b^{d/2}\right)$
- Average-Case: $\mathcal{O}\left(b^{3d/4}\right)$.

6.4 Move Ordering

Some heuristics for move ordering:

Knowledge Based Try captures first in chess.

Principal Variant (PV) A sequence of moves that programs consider best and therefore expect to be played.

Killer Move The last move at a given depth that caused $\alpha\beta$-pruning or had best minimax value.

History Table Track how often a particular move at any depth caused $\alpha\beta$-pruning or had best minimax value.

6.4.1 History Table

There are two options for history tables:

1. Generate set of legal moves and use History Table value as f value.
2. Keep moves with History Table values in a sort array and for a given state traverse the array to find the legal move with the highest History Table value.

6.5 Search Depth Heuristics

Some search depth heuristics can include:

- Time-Based/State-Based
- Horizontal Effect[3]
- Singular Extensions/Quiescence Search

When there is a time to move constraint, some plausible algorithms include:

- Constant
- Percentage of remaining time
- State dependent
- Hybrid

[3]The phenomenon of deciding on a non-optimal principal variant because an ultimately unavoidable damaging move seems to be avoided by blocking it till passed the search depth.

6.6 Quiescence Search

Quiescence states are states that could have giant swings in the heuristic values. For these states,

- When search depth reached, compute quiescence state evaluation heuristic.
- If state quiescent, then proceed as usual; otherwise increase search depth if quiescence search depth not yet reached.

6.7 Transition Tables

A transition table is a hash table of previously calculated state evaluation heuristic values. The speedup is particularly large for iterative deepening search algorithms.

7 State-Space Search

State space search is a process in which successive configurations or states of an instance are considered, with the intention of finding a goal state with a desired property. Some concepts related to state-space search are:

- Complete-state formulation
- Objective function
- Global optima
- Local optima
- Ridges, plateaus, and shoulders
- Random search and local search

Some examples are included below.

7.1 Hill Climbing

7.1.1 Steepest-Ascent Hill-Climbing

Steepest-Ascent Hill-Climbing is simply a loop that continually moves in the direction of increasing value; that is, uphill. It terminates when it reaches a "peak" where no neighbor has a higher value.

```
function HILL-CLIMBING(problem)
    current ← MAKE-NODE(problem.INITIAL-STATE)

    loop
        neighbor ← highest-valued successor of current
        if neighbor.VALUE ≤ current.VALUE then
            return current.STATE
        end if

        current ← neighbor
    end loop
end function
```

7.1.2 Stochastic Hill-Climbing

The Stochastic version is almost identical to Steepest-Ascent Hill-Climbing, except it chooses at random from among uphill moves. The probability of selection can vary with the steepness of the uphill move. It does show on average slower convergence, but also less chance of premature convergence

7.1.3 First-Choice Hill-Climbing

The First-Choice version is almost identical to Steepest-Ascent Hill-Climbing, except it chooses the first randomly generated uphill move. Although it is greedy, incomplete, and suboptimal, it's also very practical when the number of successors is large. It has an even slower convergence rate, but also a low chance of premature convergence as long as the move generation order is randomized

7.1.4 Random-Restart Hill-Climbing

The Random-Restart version is almost identical to all hill climbing, except it restarts until a goal is found. It's trivially complete. The expected number of restarts is:

$$\text{restarts} = \frac{1}{p}$$

where p is the probability of a successful hill climb given a random initial state.

7.2 Simulated-Annealing

Simulated-Annealing is hill-climbing except instead of picking the best move, it picks a random move. If the selected move improves the solution, then it is always accepted. Otherwise, the algorithm makes the move anyway with some probability less than 1. The probability decreases exponentially with the "badness" of the move, which is the amount ΔE by which the solution is worsened (i.e., energy is increased.)

```
function SIMULATED-ANNEALING(problem, schedule)
    current ← MAKE-NODE(problem.INITIAL-STATE)

    for t = 1 to ∞ do
        T ← schedule(t)
        if T = 0 then
            return current.STATE
        end if

        next ← random successor of current
        ΔE ← next.VALUE − current.VALUE
        if ΔE > 0 then
            current ← next
        else
            current ← next only with probability e^(ΔE/T)
        end if
    end for
end function
```

7.3 Population Based Local Search

7.3.1 Deterministic Local Beam Search

The local beam search is *similar* to Steepest-Ascent Hill-Climbing. The algorithm keeps track of k states rather than a single state. Although this may seem like Random-Restart Hill-Climbing with k random restarts.

In a random-restart search, each search process runs independently of the others. In a local beam search, useful information is passed among the parallel search threads. In effect, the states that generate the best successors say to the others, "Come over here, the grass is greener!" The algorithm quickly abandons unfruitful searches and moves its resources to where the most progress is being made.

In its simplest form, local beam search can suffer from a lack of diversity among the k states—they can quickly become concentrated in a small region of the state space, making the search little more than an expensive version of hill climbing.

7.3.2 Stochastic Beam Search

Instead of choosing the best k from the pool of candidate successors, stochastic beam search chooses k successors at random, with the probability of choosing a given successor being an increasing function of its value.

7.3.3 Evolutionary Algorithms

A genetic algorithm (GA) is a variant of stochastic beam search in which successor states are generated by combining two parent states rather than by modifying a single state.

```
function SIMULATED-ANNEALING(problem, FITNESS-FUNCTION)
    while some individual is fit enough, or enough tie has elapsed do
        NEW_POPULATION ← empty set

        for i = 1 to SIZE(population) do
            x ← RANDOM-SELECTION(population, FITNESS-FUNCTION)
            y ← RANDOM-SELECTION(population, FITNESS-FUNCTION)
            child ← REPRODUCE(x, y)

            if small random probability then
                child ← MUTATE(child)
            end if
            add child to new_population
        end for
    end while

    return the best individual in population, according to FITNESS-FUNCTION
end function
```

function Reproduce(x, y)
  $n \leftarrow$ Length(x)
  $c \leftarrow$ random number from 1 to n
  return Append(Substring(x, $1, c$), Substring(y, $c+1, n$))
end function

7.3.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a stochastic population-based optimization technique which assigns velocities to population members encoding trial solutions. It acts according to the rule:

$$\overrightarrow{v}^{+} = c_1 \cdot rand() \cdot (\overrightarrow{p_{\text{best}}} - \overrightarrow{p_{\text{present}}}) + c_2 \cdot rand() \cdot (\overrightarrow{g_{\text{best}}} - \overrightarrow{p_{\text{present}}})$$

and updating $\overrightarrow{p_{\text{present}}}$ as:

$$\overrightarrow{p_{\text{present}}} += \overrightarrow{v}$$

7.3.5 Ant Colony Optimization

In Ant Colony Optimization (ACO), a set of software agents called artificial ants search for good solutions to a given optimization problem. To apply ACO, the optimization problem is transformed into the problem of finding the best path on a weighted graph. The artificial ants (hereafter ants) incrementally build solutions by moving on the graph. The solution construction process is stochastic and is biased by a pheromone model, that is, a set of parameters associated with graph components (either nodes or edges) whose values are modified at runtime by the ants.

7.4 Online Search

An online search problem must be solved by an agent executing actions, rather than by pure computation. We assume a deterministic and fully observable environment (Chapter 17 relaxes these assumptions), but we stipulate that the agent knows only the following:

- Actions(s)
- c(s, a, s')[4]
- Goal-Test(s)

Online Search problems typically interleave computation and action. They environment is typically dynamic, non-deterministic, unknown domains. For the purposes of this section, we assume all exploration problems are safely explorable.

We define the following terms for Online Search Agents:

[4]This cannot be used until Result(s, a).

CR Competitive Ration

TAPC Total Actual Path Cost

C^* A Optimal Path Cost

From this, we get the Competitive Ratio as follows

$$CR = \frac{TAPC}{C^*}$$

From this, we can see we have the best case of $CR = 1$ and the worst case of $CR = \infty$.

To actually implement online agents, we introduce two new algorithms: Online-DFS-Agent and LRTA* Agent.

7.4.1 Online-DFS-Agent

```
function Online-DFS-Agent(s')
    if Goal-Test(s') then
        return stop
    end if

    if s' is a new state (not in untried) then
        untried[s'] ← Actions(s')
    end if

    if s is not null then
        result[s, a] ← s'
        add s to the front of unbacktracked[s']
    end if
    if untried[s'] is empty then
        if unbacktracked[s'] is empty then
            return stop
        else
            a ← an action b such that result[s', b] = Pop(unbacktracked[s'])
        end if
    else
        a ← Pop(untried[s'])
    end if

    s ← s'
    return a
end function
```

7.4.2 LRTA*-Agent

```
function LRTA*(s')
    if Goal-Test(s') then
        return stop
    end if

    if s' is a new state (not in H) then
        H[s'] ← h(s')
    end if

    if s is not null then
        result[s, a] ← s'
        H[s] ← min (LRTA*-Cost(s, b, result[s, b], H) ∀b ∈ Actions(s))
    end if

    a ← an action b in Actions(s') that minimizes LRTA*-Cost(s', b, result[s', b], H)
    s ← s'
    return a
end function

function LRTA*-Cost(s, a, s', H)
    if s' is undefined then
        return h(s)
    end if

    return c(s, a, s') + H[s']
end function
```

8 AI Ethics

For background, some key historical events in AI were as follows:

4th Century BC Aristotle propositional logic.

1600s Descartes mind-body connection.

1805 First programmable machine.

Mid 1800s Charles Babbage's "difference engine" and "analytical engine".

Sometime Lady Lovelace's Objection

1847 George Boole propositional logic

1879 Gottlob Frege predicate logic

1931 Kurt Godel: Incompleteness Theorem

> In any language expressive enough to describe natural number properties, there are undecidable (incomputable) true statements

1943 McCulloch & Pitts: Neural Computation

1956 Term "AI" coined

1976 Newell & Simon's "Physical Symbol System Hypothesis" A physical symbol system has the necessary and sufficient means for general intelligent action

1974–80, 1987–93 AI Winters

1980^{+} Commercialization of AI

1986^{+} Rebirth of Artificial Neural Networks

1990s Unification of Evolutionary Computation

200^{+} Rise of Deep Learning

8.1 Weak Vs. Strong AI Hypothesis

The Week AI vs. Strong AI Hypothesis is thus:

Weak AI Hypothesis It's possible to create machines that can act as if they're intelligent.

Strong AI Hypothesis It's possible to create machines that actually think.

This closely relates to the following problems:

- Rene Descartes (1596–1650)
- Rationalism, Dualism, Materialism, Brain in a Vat
- Star Trek & Souls
- Chinese Room[5]

8.2 Ethics

Some ethical concerns with regards to AIs are as follows:

- Autonomous AIs and the Trolley Dilemma
- Unemployment and Inequality
- Human dependency and obsoleteness
- Bias transfer
- Security
- Human-robot relationships
- Rights of sentient beings.

[5]The Chinese room argument holds that a program cannot give a computer a "mind", "understanding" or "consciousness", regardless of how intelligently or human-like the program may make the computer behave.

Part V

Additional Mathematics

Chapter 10

MATH 1215 - Calculus II

Chapter 6

Course:
../math1215-calculus-ii/chapter-6/chapter-6.pdf

Document 24

Included from: ../math1215-calculus-ii/chapter-6/chapter-6.pdf

00A0

Chapter 6: Applications of Integration

Illya Starikov

June 30, 2025

Contents

6.7 Physical Applications

Mass of a One-Dimensional Object

Suppose a thin bar or wire is represented by a line segment on the interval $a \leq x \leq b$ with a density function ρ (with units of mass per length). The mass of the object is

$$m = \int_a^b \rho(x)\, dx \tag{1}$$

Work

The work done by a variable force F in moving an object along a line from $x = a$ to $x = b$ in the direction of the force is

$$W = \int_a^b F(x)\, dx \tag{2}$$

Solving Lifting Problems

1. Draw a y-axis in the vertical direction (parallel to gravity) and choose a convenient origin. Assume the interval $[a, b]$ corresponds to the vertical extent of the fluid.

2. For $a \leq y \leq b$, find the cross-sectional area $A(y)$ of the horizontal slices and the distance $D(y)$ the slices must be lifted.

3. The work required to lift the water is

$$W = \int_a^b \rho g A(y) D(y)\, dy \tag{3}$$

Solving Force/Pressure Problems

1. Draw a y-axis on the face of the dam in the vertical direction and choose a convenient origin (often taken to be the base of the dam).

2. Find the width function $w(y)$ for each value of y on the face of the dam.

3. If the base of the dam is at $y = 0$ and the top of the dam is at $y = a$, then the total force on the dam is

$$F = \int \rho g(a - y)w(y)\, dy \tag{4}$$

Chapter 7

Course:
`../math1215-calculus-ii/chapter-7/chapter-7.pdf`

Document 25

Included from: `../math1215-calculus-ii/chapter-7/chapter-7.pdf`

00A0

Chapter 7: Logarithmic and Exponential Functions

Illya Starikov

June 30, 2025

Contents

7.1 Inverse Function

Derivative of the Inverse Function

Let f be differentiable and have an inverse on an interval I. If x_0 is a point of I at which $f'(x_0) \neq 0$, then f^{-1} is differentiable at $y_0 = f(x_0)$ and

$$(f^{-1})'(y_0) = \frac{1}{f'(x_0)}, \text{ where } y_0 = f(x_0) \tag{1}$$

7.2 The Natural Logarithmic and Exponential Functions

The Natural Logarithm

The **natural logarithm** of a number $x > 0$, denoted x, is defined

$$\ln x = \int_1^x \frac{dt}{t} \tag{2}$$

Properties of the Natural Logarithm

1. The domain and range of $\ln x$ are $(0, \infty)$ and $(-\infty, \infty)$, respectively.
2. $\ln(xy) = \ln x + \ln y, \forall x, y \in \mathbf{R}^+$
3. $\ln(x/y) = \ln x - \ln y, \forall x, y \in \mathbf{R}^+$
4. $\ln x^p = p \ln x, \forall x \in \mathbf{Q}^+$
5. $\frac{d}{dx}(\ln |x|) = \frac{1}{x}, \forall x \in \mathbf{R} \wedge x \neq 0$
6. $\frac{d}{dx}(\ln |u(x)|) = \frac{u'(x)}{u(x)}$
7. $\int \frac{dx}{x} = \ln |x| + C$

The Number e

The number e is the real number that satisfies

$$\ln e = \int_1^e \frac{dt}{t} = 1 \tag{3}$$

The Exponential Function

$\forall x, y \in \mathbf{R}$

$$y = e^x \Leftrightarrow x = \ln y \tag{4}$$

Properties of e

- $e^{x+y} = e^x e^y$
- $e^{x-y} = e^x / e^y$
- $(e^x)^y = e^{xy}, \forall y \in \mathbf{Q}$
- $\ln(e^x) = x, \forall x \in \mathbf{R}$
- $e^{\ln\ x} = x, \forall x \in \mathbf{R}^+$

Exponential Functions with General Bases

Let $b \in \mathbf{R}^+ \wedge b \neq 1$. $\forall x \in \mathbf{R}^+$,

$$b^x = e^{x \ln b} \tag{5}$$

Derivative and Integral of the Exponential Function

$\forall x \in \mathbf{R}$,

$$\frac{d}{dx}(e^{u(x)}) = e^{u(x)} u'(x) \tag{6}$$

$$\int e^x \, dx = e^x + C \tag{7}$$

7.3 Logarithmic and Exponential Functions with Other Bases

Logarithmic Function Base b

For any base $b > 0$, with $b \neq 1$, the **logarithmic function base** b, denoted $\log_b x$, is the inverse of the exponential function b^x.

Inverse Relations for Exponential and Logarithmic Functions

For any base $b > 0$, with $b \neq 1$, the following inverse relation holds.

- $b^{\log_b x} = x, \forall x \in \mathbf{R}^+$
- $\log_b b^x = x, \forall x$

Derivative of b^x

If $b > 0 \wedge b \neq 1, \forall x,$

$$\frac{d}{dx}(b^x) = b^x \ln b \tag{8}$$

Indefinite Integral of b^x

For $b > 0 \wedge b \neq 1,$

$$\int b^x dx = \frac{1}{\ln b} b^x + C \tag{9}$$

General Power Rule

$\forall p, x \in \mathbf{R}^+,$

$$\frac{d}{dx}(x^p) = px^{p-1} \tag{10}$$

Furthermore, if u is a positive differentiable function on its domain, then

$$\frac{d}{dx}(u(x)^p) = p(u(x))^{p-1} \times u'(x) \tag{11}$$

Derivative of $\log_b x$

If $b > 1$,

$$\frac{d}{dx}(\log_b x) = \frac{1}{x \ln b} \quad \forall x \in \mathbf{R}^+ \wedge x \neq 0 \tag{12}$$

$$\frac{d}{dx}(\log_b |x|) = \frac{1}{x \ln b} \quad \forall x \in \mathbf{R}^+ \wedge x \neq 0 \tag{13}$$

7.5 Inverse Trigonometric Functions

Derivative of Inverse Trigonometric Functions

$\frac{d}{dx}(\sin^{-1} x) = \frac{1}{\sqrt{1-x^2}}$ $\quad \frac{d}{dx}(\cos^{-1} x) = -\frac{1}{\sqrt{1-x^2}}$, $\{x \in \mathbf{R} \mid -1 < x < 1\}$

$\frac{d}{dx}(\tan^{-1} x) = \frac{1}{1+x^2}$ $\quad \frac{d}{dx}(\cot^{-1} x) = -\frac{1}{1+x^2}$, $\{x \in \mathbf{R} \mid -\infty < x < \infty\}$

$\frac{d}{dx}(\sec^{-1} x) = \frac{1}{|x|\sqrt{x^2-1}}$ $\quad \frac{d}{dx}(\csc^{-1} x) = -\frac{1}{|x|\sqrt{x^2-1}}$, $\{x \in \mathbf{R} \mid |x| > 1\}$

Integrals Involving Inverse Trigonometric Functions

$$\int \frac{dx}{\sqrt{a^2 - x^2}} = \sin^{-1} \frac{x}{a} + C \tag{14}$$

$$\int \frac{dx}{a^2 + x^2} = \frac{1}{a} \tan^{-1} \frac{x}{a} + C \tag{15}$$

$$\int \frac{dx}{x\sqrt{a^2 + x^2}} = \frac{1}{a} \sec^{-1} \left|\frac{x}{a}\right| + C \tag{16}$$

7.6 L'Hôpital's Rule and Growth Rates of Functions

Indeterminate forms $1^\infty, 0^0, \infty^0$

Assume $\lim\limits_{x\to a} f(x)^{g(x)}$ has the indeterminate form 1^∞, 0^0, or ∞^0.

1. Evaluate $L = \lim\limits_{x\to a} g(x)\ln f(x)$. This limit can be put in the form $0/0$ or ∞/∞, both of which are handled by l'Hôpital's rule.

2. Then $\lim\limits_{x\to a} f(x)^{g(x)} = e^L$

Growth Rates of Functions (as $x \to \infty$)

Suppose f and g are functions with $\lim\limits_{x\to\infty} f(x) = \lim\limits_{x\to\infty} g(x) = \infty$. Then f **grows faster than g** as $x \to \infty$ if

$$\lim_{x\to\infty} \frac{g(x)}{f(x)} = 0 \quad \text{or, quantitatively, if} \quad \lim_{x\to\infty} \frac{f(x)}{g(x)} = \infty \tag{17}$$

The functions f and g have **comparable growth rates** if

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = M$$

where $M \in \mathbb{R}^+$.

Ranking Growth Rates as $x \to \infty$

Let $f << g$ mean that g grows faster than f as $f \to \infty$. With positive real numbers p, q, r, s and $b > 1$,

$$\ln^q x << x^p << x^p \ln^r x << x^{p+s} << b^x << x^x \tag{18}$$

7.7 Hyperbolic Functions

Hyperbolic Functions

Hyperbolic Cosine

$$\cosh x = \frac{e^x + e^{-x}}{2} \tag{19}$$

Hyperbolic Sine

$$\sinh x = \frac{e^x - e^{-x}}{2} \tag{20}$$

Hyperbolic Tangent

$$\tanh x = \frac{\sinh x}{\cosh x} = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{21}$$

Hyperbolic Cotangent

$$\coth x = \frac{\cosh x}{\sinh x} = \frac{e^x + e^{-x}}{e^x - e^{-x}} \tag{22}$$

Hyperbolic Secant

$$\operatorname{sech} x = \frac{1}{\cosh x} = \frac{2}{e^x + e^{-x}} \tag{23}$$

Hyperbolic Cosecant

$$\operatorname{csch} x = \frac{1}{\sinh x} = \frac{2}{e^x - e^{-x}} \tag{24}$$

Hyperbolic Identities

$\cosh^2 x - \sinh^2 x = 1$	$\cosh(-x) = \cosh x$
$1 - \tanh^2 x = \operatorname{sech}^2 x$	$\sinh(-x) = -\sinh x$
$\coth^2 x - 1 = \operatorname{csch}^2 x$	$\tanh(-x) = -\tanh x$
$\coth(x + y) = \cosh x \cosh y + \sinh x \sinh y$	
$\sinh(x + y) = \sinh x \cosh y + \cosh x \sinh y$	
$\cosh 2x = \cosh^x + \sinh^2$	$\sinh 2x = 2 \sinh x \cosh x$
$\cosh^2 x = \frac{\cosh 2x + 1}{2}$	$\sinh^2 x = \frac{\cosh 2x - 1}{2}$

Derivatives and Integral Formulas

1. $\frac{d}{dx}(\cosh x) = \sinh x \quad \Rightarrow \quad \int \sinh x \; dx = \cosh x + C$
2. $\frac{d}{dx}(\sinh x) = \cosh x \quad \Rightarrow \quad \int \cosh x \; dx = \sinh x + C$
3. $\frac{d}{dx}(\tanh x) = \operatorname{sech}^2 x \quad \Rightarrow \quad \int \operatorname{sech}^2 x \; dx = \tanh x + C$
4. $\frac{d}{dx}(\coth x) = -\operatorname{csch}^2 x \quad \Rightarrow \quad \int \operatorname{csch}^2 x \; dx = -\coth x + C$
5. $\frac{d}{dx}(\operatorname{sech} x) = -\operatorname{sech} x \tanh x \quad \Rightarrow \quad \int \operatorname{sech} x \tanh x \; dx = -\operatorname{sech} x + C$
6. $\frac{d}{dx}(\operatorname{csch} x) = -\operatorname{csch} x \coth x \quad \Rightarrow \quad \int \operatorname{csch} x \coth x \; dx = -\operatorname{csch} x + C$

Integrals of Hyperbolic Functions

1. $\int \tanh x \, dx = \ln \cosh x + C$
2. $\int \coth x \, dx = \ln|\sinh x| + C$
3. $\int \operatorname{sech} x \, dx = \tan^{-1}|\sinh x| + C$
4. $\int \operatorname{csch} x \, dx = \ln|\tanh(x/2)| + C$

Inverses of the Hyperbolic Functions Expressed as Logarithms

$\cosh^{-1} x = \ln(x + \sqrt{x^2 - 1}) \ (x \geq 1)$ $\quad \operatorname{sech}^{-1} x = \cosh^{-1} \frac{1}{x} \ (0 < x \leq 1)$
$\sinh^{-1} x = \ln(x + \sqrt{x^2 + 1})$ $\quad \operatorname{csch}^{-1} x = \sinh^{-1} \frac{1}{x} \ (x \neq 0)$
$\tanh^{-1} x = \frac{1}{2} \ln(\frac{1+x}{1-x}) \ (|x| < 1)$ $\quad \coth^{-1} x = \tanh^{-1} \frac{1}{x} \ (|x| > 1)$

Derivatives of the Inverse Hyperbolic Functions

$\frac{d}{dx}(\cosh^{-1} x) = \frac{1}{\sqrt{x^2-1}} \ (x > 1)$ $\quad \frac{d}{dx}(\sinh^{-1} x) = \frac{1}{\sqrt{x^2+1}}$
$\frac{d}{dx}(\tanh^{-1} x) = \frac{1}{1-x^2} \ (|x| < 1)$ $\quad \frac{d}{dx}(\coth^{-1} x) = \frac{1}{1-x^2} \ (|x| > 1)$
$\frac{d}{dx}(\operatorname{sech}^{-1} x) = -\frac{1}{x\sqrt{1-x^2}} \ (0 < x < 1)$ $\quad \frac{d}{dx}(\operatorname{csch}^{-1} x) = -\frac{1}{|x|\sqrt{1+x^2}} \ (x \neq 0)$

Integral Formulas

1. $\int \frac{dx}{\sqrt{x^2-a^2}} = \cosh^{-1} \frac{x}{a} + C, \text{ for } x > a$
2. $\int \frac{dx}{x^2+a^2} = \sinh^{-1} \frac{x}{a} + C, \text{ for all } x$

3. $\int \frac{dx}{a^2-x^2} = \frac{1}{a}\tanh^{-1}\frac{x}{a} + C,\ \text{for} |x| < a = \frac{1}{a}\coth^{-1}\frac{x}{a} + C, \text{for } |x| > a$

4. $\int \frac{dx}{x\sqrt{a^2-x^2}} = -\frac{1}{a}\operatorname{sech}^{-1}\frac{x}{a} + C,\ \text{for } 0 < x < a$

5. $\int \frac{dx}{x\sqrt{a^2+x^2}} = -\frac{1}{a}\operatorname{csch}^{-1}\frac{|x|}{a} + C,\ \text{for } x \neq 0$

Chapter 8

Course:
../math1215-calculus-ii/chapter-8/chapter-8.pdf

Document 26

Included from: ../math1215-calculus-ii/chapter-8/chapter-8.pdf

00A0

Chapter 8: Integration Techniques

Illya Starikov

June 30, 2025

Contents

8.1 Basic Approaches

$$\int k\,dx = kx + C \tag{1}$$

$$\int k^p\,dx = \frac{k^{p+1}}{p+1} + C, p \in \mathbf{R} \wedge \neq -1 \tag{2}$$

$$\int \cos ax\,dx = \frac{1}{a}\sin ax + C \tag{3}$$

$$\int \sin ax\,dx = -\frac{1}{a}\cos ax + C \tag{4}$$

$$\int \sec^2 ax\,dx = \frac{1}{a}\tan ax + C \tag{5}$$

$$\int \csc^2 ax\,dx = -\frac{1}{a}\cot ax + C \tag{6}$$

$$\int \sec ax \tan ax\,dx = \frac{1}{a}\sec ax + C \tag{7}$$

$$\int \csc ax \cot ax\,dx = -\frac{1}{a}\csc ax \cot ax + C \tag{8}$$

$$\int e^{ax}\,dx = \frac{1}{a}e^{ax} + C \tag{9}$$

$$\int \frac{dx}{x} = \ln|x| + C \tag{10}$$

$$\int \frac{dx}{\sqrt{a^2 - x^2}} = \sin^{-1}\frac{x}{a} + C \tag{11}$$

$$\int \frac{dx}{a^2 + x^2} = \frac{1}{a}\tan^{-1}\frac{x}{a} + C \tag{12}$$

$$\int \frac{dx}{x\sqrt{x^2 - a^2}} = \frac{1}{a}\sec^{-1}\left|\frac{x}{a}\right| + C \tag{13}$$

8.2 Integration By Parts

Suppose that u and v are differentiable functions. Then

$$\int u\,dv = uv - \int v\,du \tag{14}$$

Integration by Parts for Definite Integrals

Let u and v be differentiable. Then

$$\int_a^b u(x)v'(x)\,dx = u(x)v(x)\Big|_a^b - \int_a^b v(x)u'(x)\,dx \tag{15}$$

Integral of $\ln x$

$$\int \ln x\,dx = x\ln x - x + C \tag{16}$$

8.3 Trigonometric Integrals

$\int \sin^m x \cos^n x\ dx$ **Strategy**

m **is odd,** n **real** Split off $\sin x$, rewrite the resulting even power of $\sin x$ in terms of $\cos x$, and then use $u = \cos x$

n **odd,** m **real** Split off $\cos x$, rewrite the resulting even power of $\cos x$ in terms of $\sin x$, and then use $u = \sin x$.

m **and** n **both even, nonnegative** Use half-angle identities to transform the integrand into a polynomial in $\cos 2x$, and apply the preceding strategies once again to powers of $\cos 2x$ greater than 1.

Reduction Formulas

Assume n is a positive integer.

$$\int \sin^n x\,dx = -\frac{\sin^{n-1} x \cos x}{n} + \frac{n-1}{n} \int \sin^{n-2} x\ dx \tag{17}$$

$$\int \cos^n x\,dx = \frac{\cos^{n-1} x \sin x}{n} + \frac{n-1}{n} \int \cos^{n-2} x\ dx \tag{18}$$

$$\int \tan^n x\,dx = \frac{\tan^{n-1} x}{n-1} - \int \tan^{n-2} x\ dx, n \neq 1 \tag{19}$$

$$\int \sec^n x\,dx = \frac{\sec^{n-2} x \tan x}{n-1} + \frac{n-2}{n-1} \int \sec^{n-2} x\ dx, n \neq 1 \tag{20}$$

Integrals of $\tan x, \cot x, \sec x,$ **and** $\csc x$

$$\int \tan x\,dx = -\ln|\cos x| + C = \ln|\sec x| + C \tag{21}$$

$$\int \cot x\,dx = \ln|\sin x| + C \tag{22}$$

$$\int \sec x\,dx = \ln|\sec x + \tan x| + C \tag{23}$$

$$\int \csc x\,dx = -\ln|\csc x + \cot x| + C \tag{24}$$

$\int \tan^m x \sec^n x \, dx$ **Strategy**

n **even** Split off $\sec^2 x$, rewrite the remaining even power of $\sec x$ in terms of $\tan x$, and use $u = \tan x$.

m **odd** Split off $\sec x \tan x$, rewrite the remaining even power of $\tan x$ in terms of $\sec x$, and use $u = \sec x$.

m **even and** n **odd** Rewrite the even power of $\tan x$ in terms of $\sec x$ to produce a polynomial in $\sec x$; apply reduction formula 4 to each term.

8.4 Trigonometric Substitutions

The Integral Contains...	**Corresponding Substitution**	**Useful Identity**
$a^2 - x^2$	$x = a\sin\theta, -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}, \forall \lvert x\rvert \leq a$	$a^2 - a^2\sin^2\theta = a^2\cos^2\theta$
$a^2 + x^2$	$x = a\tan\theta, -\frac{\pi}{2} < \theta < \frac{\pi}{2}$	$a^2 + a^2\tan^2\theta = a^2\sec^2\theta$
$x^2 - a^2$	$x = a\sec\theta,$ $\begin{cases} 0 \leq \theta < \frac{\pi}{2}, \forall x \geq a \\ \frac{\pi}{2} < \theta \leq \pi, \forall x \leq -a \end{cases}$	$a^2\sec^2\theta - a^2 = a^2\tan^2\theta$

8.5 Partial Fractions

Partial Fractions with Simple Linear Factors

Suppose $f(x) = p(x) > q(x)$, where p and q are polynomials with no common factors and with the degree of p less than the degree of q. Assume that q is the product of simple linear factors. The partial fraction decomposition is obtained as follows.

Step 1. Factor the denominator q in the form $(x-r_1)(x-r_2)\cdots(x-r_n)$, where $r_1, \ldots, r_n$ are real numbers.

Step 2. Partial fraction decomposition Form the partial fraction decomposition by writing

$$\frac{p(x)}{q(x)} = \frac{A_1}{x-r_1} + \frac{A_2}{x-r_2} + \cdots + \frac{A_n}{x-r_n} \tag{25}$$

Step 3. Clear denominators Multiply both sides of the equation in Step 2 by $q(x) = (x-r_1)(x-r_2)\cdots(x-r_n)$, which produces conditions for $A_1, \ldots, A_n$.

Step 4. Solve for coefficients Equate like powers of x in Step 3 to solve for the undertmined coefficients $A_1, \ldots, A_n$

Partial Fractions For Repeated Linear Factors

Suppose the repeated linear factor $(x-r)^m$ appears in the denominator of a proper rational function in reduced form. The partial fraction decomposition has a partial fraction for each power of $(x-r)$ up to and including the mth power; that is, the partial fraction decomposition contains the sum

$$\frac{A_1}{(x-r)} + \frac{A_2}{(x-r)^2} + \frac{A_3}{(x-r)^3} + \cdots + \frac{A_m}{(x-r)^m} \tag{26}$$

where $A_1, \ldots, A_m$ are constants to be determined.

Partial Fractions with Simple Irreducible Quadratic Factors

Suppose a simple irreducible factor $ax^2 + bx + c$ appears in the denominator of a proper rational function in reduced form. The partial fraction decomposition contains a term of the form

$$\frac{Ax+B}{ax^2+bx+c} \tag{27}$$

where A and B are unknown coefficients to be determined.

Partial Fraction Decomposition

Let $f(x) = p(x)/q(x)$ be a proper rational function in reduced form. Assume the denominator q has been factored completely over the real numbers and m is a positive integer.

Simple linear factor A factor $x-r$ in the denominator requires the partial fraction $\frac{A}{x-r}$.

Repeated linear factor A factor $(x-r)^m$ with $m > 1$ in the denominator requires the partial fractions.

$$\frac{A_1}{(x-r)} + \frac{A_2}{(x-r)^2} + \frac{A_3}{(x-r)^3} + \cdots + \frac{A_m}{(x-r)^m} \tag{28}$$

Simple irreducible quadratic factor An irreducible factor ax^2+bx+c in the denominator requires the partial fraction

$$\frac{Ax+B}{ax^2+bx+c} \tag{29}$$

Repeated irreducible quadratic factor An irreducible factor $(ax^2+bx+c)^m$ with $m > 1$ in the denominator requires the partial fractions

$$\frac{A_1x+B_1}{ax^2+bx+c} + \frac{A_2x+B_2}{(ax^2+bx+c)^2} + \cdots + \frac{A_mx+B_m}{(ax^2+bx+c)^m} \tag{30}$$

8.8 Improper Integrals

Improper Integrals over Infinite Intervals

1. If f is continuous on $[a, \infty)$, then

$$\int_a^\infty f(x)\,dx = \lim_{b\to\infty} \int_a^b f(x)\,dx \tag{31}$$

provided the limit exists.

2. If f is continuous on $(-\infty, b]$, then

$$\int_{-\infty}^b f(x)\,dx = \lim_{a\to-\infty} \int_a^b f(x)\,dx \tag{32}$$

provided the limit exists.

3. If f is continuous on $(-\infty, \infty)$, then

$$\int_{-\infty}^\infty f(x)\,dx = \lim_{a\to-\infty} \int_a^c f(x)\,dx + \lim_{b\to\infty} \int_c^b f(x)\,dx \tag{33}$$

provided both limits exist, where c is any real number.

In each case, if the limit exists, the improper integral is said to **converge**, if it does not exist, the improper integral is said to **diverge**.

Improper Integrals with an Unbounded Integrand

1. Suppose f is continuous on $(a, b]$ with $\lim_{x\to a^+} f(x) = \pm\,\infty$. Then

$$\int_a^b f(x)\,dx = \lim_{c\to a^+} \int_c^b f(x)\,dx \tag{34}$$

provided the limit exists.

2. Suppose f is continuous on $[a, b)$ with $\lim_{x\to b^-} f(x) = \pm\,\infty$. Then

$$\int_a^b f(x)\,dx = \lim_{c\to b^-} \int_a^c f(x)\,dx \tag{35}$$

provided the limit exists.

3. Suppose f is continuous on $[a, b]$ except at the interior point p where f is unbounded. Then

$$\int_a^b f(x)\,dx = \int_a^p f(x)\,dx + \int_p^b f(x)\,dx, \tag{36}$$

provided the improper integrals on the right side exist.

In each case, if the limit exists, the improper integrals is said to **converge**, if it does not exists, the improper integral is said to **diverge**.

Chapter 9

Course:
../math1215-calculus-ii/chapter-9/chapter-9.pdf

Document 27

Included from: ../math1215-calculus-ii/chapter-9/chapter-9.pdf

00A0

Chapter 9: Sequences and Infinite Series

Illya Starikov

June 30, 2025

Contents

9.1 An Overview

Sequence

A **sequence** $\{a_n\}$ is an ordered list of numbers of the form

$$\{a_1, a_2, a_3, \ldots, a_n, \ldots\} \tag{1}$$

A sequence may be generated by a **recurrence relations** of the form $a_{n+1} = f(a_n)$, for $n = 1, 2, 3, \ldots$, where a_1 is given. A sequence may also be defined with an **explicit form** of the form $a_n = f(n)$, for $n = 1, 2, 3, \ldots$.

Limit of a Sequence

If the terms of a sequence $\{a_n\}$ approach a unique number L as n increases, then we say $\lim\limits_{n\to\infty} = L$ exists, and the sequence **converges** to L. If the terms of the sequence do not approach a single number as n increases, the sequence has no limits, and the sequence **diverges**.

Infinite Series

Given a set of numbers $\{a_1, a_2, a_3, \ldots\}$, the sum

$$a_1 + a_2 + a_3 + \cdots = \sum_{k=1}^{\infty} a_k \tag{2}$$

is called an **infinity series**. Its **sequence of partial sums** $\{S_n\}$ has the terms

$$S_1 = a_1 \tag{3}$$

$$S_2 = a_1 + a_2 \tag{4}$$

$$S_3 = a_2 + a_2 + a_3 \tag{5}$$

$$\vdots \tag{6}$$

$$S_n = a_1 + a_2 + a_3 + \cdots + a_n = \sum_{k=1}^{n} a_k, \quad n = 1, 2, 3, \ldots \tag{7}$$

If the sequence of partial sums $\{S_n\}$ has a limit L, the infinite series **converges** to that limits, and we write

$$\sum_{k=1}^{\infty} a_k = \lim_{n\to\infty} \sum_{k=1}^{n} a_k = \lim_{n\to\infty} S_n = L \tag{8}$$

If the sequence of partial sums diverges, the infinite series also **diverges**.

9.2 Sequences

Limits of Sequences from Limits of Functions

Suppose f is a function such that $f(n) = a_n$ for all positive integers n. If $\lim_{x\to\infty} = L$, then the limits of the sequences $\{a_n\}$ is also L.

Properties of Limits of Sequences

Assume that the sequence $\{a_n\}$ and $\{b_n\}$ have limits A and B, respectively. Then,

1. $\lim_{n\to\infty}(a_n \pm b_n) = A \pm B$
2. $\lim_{n\to\infty} ca_n = cA$, where c is a real number
3. $\lim_{n\to\infty} a_n b_n = AB$
4. $\lim_{n\to\infty} \frac{a_n}{b_n} = \frac{A}{B}$, provided $B \neq 0$.

Geometric Sequences

Let r be a real number. Then,

$$\lim_{n\to\infty} r^n = \begin{cases} 0 & \Leftrightarrow |r| < 1 \\ 1 & \Leftrightarrow r = 1 \\ \text{does not exist} & \Leftrightarrow r \leq -1 \vee r > 1 \end{cases} \tag{9}$$

Squeeze Theorem for Sequences

Let $\{a_n\}$, $\{b_n\}$, and $\{c_n\}$ be sequences with $a_n \leq b_n \leq c_n$ for all n greater than some index N. If $\lim_{n\to\infty} a_n = \lim_{n\to\infty} c_n = L$, the $\lim_{n\to\infty} b_n = L$.

Bounded Monotonic Sequences

A bounded monotonic sequence converges.

Growth Rates of Sequences

The following sequences are ordered according to increasing growth rates as $n \to \infty$; that is, if $\{a_n\}$ appears before $\{b_n\}$ in the list, then $\lim_{n\to\infty} \frac{a_n}{b_n} = 0$ and $\lim_{n\to\infty} \frac{b_n}{a_n} = \infty$

$$\{\ln^q n\} \ll \{n^p\} \ll \{n^p \ln^r n\} \ll \{n^{p+s}\} \ll \{b^n\} \ll \{n!\} \ll \{n^n\} \tag{10}$$

The ordering applies for $p, q, r, s, b \in \mathbb{R}^+ \wedge b > 1$.

Limit of a Sequence

The sequence $\{a_n\}$ converges to L provided the terms of a_n can be made arbitrarily close to L by taking n sufficiently large. More precisely, $\{a_n\}$ has the unique limit L if given any tolerance $\epsilon > 0$, it is possible to find a positive integer N (depending only on ϵ) such that

$$|a_n - L| < \epsilon \quad \text{whenever } n > N \tag{11}$$

if the **limit of a sequence** is L, we say the sequence **converges** to L, written

$$\lim_{n\to\infty} a_n = L \tag{12}$$

A sequence that does not converge is said to **diverge.**

9.3 Infinite Series

Geometric Series

$$\sum_{k=0}^{n-1} ar^k = S_n = a\frac{1-r^n}{1-r} \tag{13}$$

Geometric Series

Let $a \neq 0$ and r be real numbers. If $|r| < 1$, then $\sum_{k=0}^{\infty} ar^k = \frac{a}{1-r}$. If $|r| \geq 1$, then the series diverges.

9.4 The Divergence and Integral Tests

Divergence Test

If $\sum a_k$ converges, then $\lim\limits_{k\to\infty} a_k = 0$. Equivalently, if $\lim\limits_{k\to\infty} a_k \neq 0$, then the series diverges. However, this cannot be used to prove convergence. If $\lim\limits_{k\to\infty} a_k = 0$, the test is inconclusive.

Harmonic Series

The harmonic series $\sum_{k=1}^{\infty} \frac{1}{k} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \cdots$ diverges, even though the terms of the series approach zero.

Integral Test

Suppose f is a continuous, positive, decreasing function, for $x \geq 1$, and let $a_k = f(k)$, for $k = 1, 2, 3, \ldots$. Then

$$\sum_{k=1}^{\infty} a_k \quad \text{and} \quad \int_1^{\infty} f(x)\,dx \tag{14}$$

either both converge or both diverge. In the case of convergence, the value of the integral is *not*, in general, equal to the value of the series.

Convergence of the p-Series

The p-Series $\sum_{k=1}^{\infty} \frac{1}{k^p}$ converges, for $p > 1$, and diverges for $p \leq 1$.

Estimating Series with Positive Terms

Let f be continuous, positive, decreasing function, for $x \geq 1$, and let $a_k = f(k)$, for $k = 1, 2, 3, \ldots$. Let $S = \sum_{k=1}^{\infty} a_k$ be a convergence series and let $S_n = \sum_{k=1}^{n}$ be the sum of of the first n terms of the series. The remainder $R_n = S - S_n$ satisfies

$$R_n \leq \int_n^{\infty} f(x)\,dx \tag{15}$$

Furthermore, the exact value of the series is bounded as follows:

$$S_n + \int_{n+1}^{\infty} f(x)\,dx \leq \sum_{k=1}^{\infty} a_k \leq S_n + \int_n^{\infty} f(x)\,dx. \tag{16}$$

Properties of Convergent Series

1. Suppose $\sum a_k$ converges to A and let c be a real number. The series $\sum ca_k$ converges and $\sum ca_k = c \sum a_k = cA$

2. Suppose $\sum a_k$ converges to A and $\sum b_k$ converges to B. The series $\sum(a_k \pm b_k)$ converges and $\sum(a_k \pm b_k) = \sum a_k \pm \sum b_k = A \pm B$

3. *Whether* a series converges does not depend on a fininte number of terms added to or removed form the series. Specifically, if M is a positive integer, then $\sum_{k=1}^{\infty} a_k$ and $\sum_{k=M}^{\infty} a_k$ both converge or both diverge. However, the *value* of a convergent series does change if nonzero terms are added or deleted.

9.5 The Ratio, Root, and Comparison Tests

Useful Identities

$$\lim_{k\to\infty}\left(1+\frac{1}{k}\right)^k = \lim_{k\to\infty}\left(\frac{k+1}{k}\right)^k = e \tag{17}$$

$$\lim_{k\to\infty} k^{\frac{1}{k}} = 1 \tag{18}$$

The Ratio Test

Let $\sum a_k$ be an infinite series with positive terms and let $r = \lim_{k\to\infty} \frac{a_{k+1}}{a_k}$.

1. If $0 \leq r < 1$, the series converges.
2. If $r > 1$ (including $r = \infty$), the series diverges.
3. If $r = 1$, the test is inconclusive.

The Root Test

Let $\sum a_k$ be an infinite series with nonegative terms and let $\rho = \lim_{k\to\infty} \sqrt[k]{a_k}$.

1. If $0 \leq \rho < 1$, the series converges.
2. If $\rho > 1$ (including $\rho = \infty$), the series diverges.
3. If $\rho = 1$, test is inconclusive.

The Comparison Test

Let $\sum a_k$ and $\sum b_k$ be a series with postivie terms.

1. If $0 < a_k \leq b_k$ and $\sum b_k$ converges, then $\sum a_k$ converges.
2. If $0 < b_k \leq a_k$ and $\sum b_k$ diverges, then $\sum a_k$ diverges.

The Limit Comparison Test

Let $\sum a_k$ and $\sum b_k$ be series with positive terms and let

$$\lim_{k\to\infty} \frac{a_k}{b_k} = L \tag{19}$$

- If $0 < L < \infty$ (that is, L is a finite, positive number), then $\sum a_k$ and $\sum b_k$ either both converge or both diverge.
- If $L = 0$ and $\sum b_k$ converges, then $\sum a_k$ converges.
- If $L = \infty$ and $\sum b_k$ diverges, then $\sum a_k$ diverges.

Guidelines

- Begin with the Divergence Test. If you show that $\lim_{k\to\infty} a_k \neq 0$, then the series diverges and your work is finished.
- Is the series a special series? Recall the convergence properties for the following series:
  - Geometric series: $\sum ar^k$ converges for $|r| < 1$ and diverges for $|r| \geq 1$ $(a \neq 0)$.
  - p-series: $\sum \frac{1}{k^p}$ converges for $p > 1$, and diverges for $p \leq 1$.
  - Check also for telescoping series.
- If the general kth term of the series looks like a function you can integrate, then try the Integral Test.
- If the general kth term of the series involves $k!, k^k,$ or a^k, where a is a constant, the Ratio Test is advisable. Series with k in an exponent may yield to the Root Test.
- If the general kth term of the series is a rational function of k (or a root of a rational function), use the Comparison or the Limit Comparison Test. Use the families of series given in Step 2 as comparison series.

9.6 Alternating Series

The Alternating Series Test

The alternating series $\sum(-1)^{k+1}a_k$ converges provided

1. the terms of the series are non-increasing in magnitude ($0 < a_{k+1} \leq a_k$, for k greater than some index N) and

2. $\lim\limits_{k\to\infty} \quad = 0$

Alternating Harmonic Series

The alternating harmonic series $\sum_{k=1}^{\infty} \frac{(-1)^{k+1}}{k} = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} - \cdots$ converges (even though the harmonic series $\sum_{k=1}^{\infty} \frac{1}{k} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \cdots$ diverges).

Remainder in Alternating Series

Let $R_n = |S - S_n|$ be the remainder in approximating the value of a convergent alternating series $\sum_{k=1}^{\infty}(-1)^{k+1}a_k$ by the sum of its first n terms. Then $R_n \leq a_{n+1}$. In other words, the remainder is less than or equal to the magnitude of the first neglected term.

Absolute and Conditional Convergence

Assume the infinite series $\sum a_k$ converges. The series $\sum a_k$ **converges absolutely** if the series $\sum |a_k|$ converges. Otherwise, the series $\sum a_k$ **converges conditionally**.

Absolute Convergence Implies Convergence

If $\sum |a_k|$ converges, then $\sum a_k$ converges (absolute convergence implies convergence). If $\sum a_k$ diverges, then $\sum |a_k|$ diverges.

Series or Test	Form of Series	Condition for Convergence	Condition for Divergence	Comments
Geometric Series	$\sum_{k=0}^{\infty} ar^k, a \neq 0$	$\|r\| < 1$	$\|r\| \geq 1$	If $\|r\| < 1$, then $\sum a_{k=1}^{\infty} ar^k = \frac{a}{1-r}$
Divergence Test	$\sum_{k=1}^{\infty} a_k$	Does Not Apply	$\lim_{k\to\infty a_k \neq 0}$	Cannot be used to prove convergence.
Integral Test	$\sum a_{k=1}^{\infty} a_k$, where $a_k = f(k)$ and f is continuous, positive, and decreasing.	$\int_1^{\infty} f(x)dx < \infty$	$\int_1^{\infty} f(x)dx$ does not exist	The value of the integral is not the value of the series.
p-Series	$\sum a_{k=1}^{\infty} \frac{1}{k^p}$	$p > 1$	$p \leq 1$	Useful for comparison tests.
Ratio Test	$\sum a_{k=1}^{\infty}$ where $a_k > 0$	$\lim_k \to \infty \frac{a_{k+1}}{a_k} < 1$	$\lim_{k\to\infty} \frac{a_{k+1}}{a_k} > 1$	Inconclusive if $\lim_{k\to\infty} \frac{a_{k+1}}{a_k} = 1$
Root Test	$\sum a_{k=1}^{\infty}$ where $a_k \geq 0$	$\lim_{k\to\infty} \sqrt[k]{a_k} < 1$	$\lim_{k\to\infty} \sqrt[k]{a_k} > 1$	Inconclusive if $\lim_{k\to\infty} \sqrt[k]{a_k} = 1$
Comparison Test	$\sum a_{k=1}^{\infty}$ where $a_k > 0$	$0 < a_k \leq b_k$ and $\sum_{k=1}^{\infty} b_k$ converges	$0 < b_k \leq a_k$ and $\sum a_{k=1}^{\infty} b_k$ diverges	$\sum a_{k=1}^{\infty} a_k$ is given; you supply $\sum a_{k=1}^{\infty} b_k$
Limit Comparison Test	$\sum a_{k=1}^{\infty} a_k$ where $a_k > 0, b_k > 0$	$0 \leq \lim_{k\to\infty} \frac{a_k}{b_k} < \infty$ and $\sum_{k=1}^{\infty} b_k$ converges	$\lim_{k\to\infty} \frac{a_k}{b_k} > 0$ and $\sum a_{k=1}^{\infty} b_k$ diverges.	$\sum a_{k=1}^{\infty} a_k$ is given; you supply $\sum a_{k=1}^{\infty} b_k$
Alternating Series Test	$\sum_{k=1}^{\infty} (-1)^k a_k$, where $a_k > 0, 0 < a_{k+1} \leq a_k$	$\lim_{k\to\infty} a_k = 0$	$\lim_{k\to\infty} a_k \neq 0$	Remainder R_n satisfies $R_n \leq a_{n+1}$
Absolute Convergence	$\sum a_{k=1}^{\infty} a_k, a_k$ arbitrary	$\sum_{k=1}^{\infty} \|a_k\|$ converges.	Applies to arbitrary series	

Chapter 10

Course:
../math1215-calculus-ii/chapter-10/chapter-10.pdf

Document 28

Included from: ../math1215-calculus-ii/chapter-10/chapter-10.pdf

00A0

Chapter 10: Power Series

Illya Starikov

June 30, 2025

Contents

10.1 Approximating Functions with Polynomials

Taylor Polynomials

Let f be a function with $f', f'', \ldots, f^{(n)}$ defined at a. The **nth-order Taylor polynomial** for f with its **center** at a, denoted p_n, has the property that it matches f in value, slope, and all derivatives up to the nth derivative at a; that is,

$$p_n(a) = f(a), p_n'(a) = f'(a), \ldots, p_n^{(n)}(a) = f^{(n)}(a). \tag{1}$$

The nth-order Taylor polynomial centered at a is

$$p_n(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(n)}(a)}{n!}(x-a)^n. \tag{2}$$

More compactly, $p_n(x) = \sum_{k=0}^{n} c_k(x-a)^k$, where the **coefficients** are

$$c_k = \frac{f^{(k)}(a)}{k!}, \qquad \text{for } k = 0, 1, 2, \ldots, n \tag{3}$$

Remainder in a Taylor Polynomial

Let p_n be the Taylor polynomial of order n for f. The **remainder** in using p_n to approximate f at the point x is

$$R_n(x) = f(x) - p_n(x) \tag{4}$$

Taylor's Theorem

Let f have continuous derivatives up to $f^{(n+1)}$ on an open interval I containing a. For all x in I,

$$f(x) = p_n(x) + R_n(x), \tag{5}$$

where p_n is the nth-order Taylor polynomial for f centered at a, and the remainder is

$$R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}(x-a)^{(n+1)}, \tag{6}$$

for some point c between x and a.

Estimate of the Remainder

Let n be a fixed positive integer. Suppose there exists a number M such that $|f^{(n+1)}(c)| \leq M$, for all c between a and x inclusive. The remainder in the nth-order Taylor polynomial for f centered at a satisfies

$$|R_n(x)| = |f(x) - p_n(x)| \leq M \frac{|x-a|^{n+1}}{(n+1)!} \tag{7}$$

10.2 Properties of Power Series

Power Series

A **power series** has the general form

$$\sum_{k=0}^{\infty} c_k (x-a)^k \tag{8}$$

where a and c_k are real numbers, and x is a variable. The c_k's are the **coefficients** of the power series and a is the **center** of the power series. The set of values of x for which the series converges is its **interval of convergence**. The **radius of convergence** of the power series, denoted R, is the distance from the center of the series to the boundary of the interval of convergence.

Convergence of Power Series

A power series $\sum_{k=0}^{\infty} c_k (x-a)^k$ centered at a converges in one of three ways:

1. The series converges absolutely for all x, in which case the interval of convergence is $(-\infty, \infty)$ and the radius of convergence is $R = \infty$.

2. There is a real number $R > 0$ such that the series converges absolutely for $|x-a| < R$ and diverges for $|x-a| > R$, in which case the radius of convergence is $R = 0$.

3. The series converges only at a, in which case the radius of convergence is $R = 0$.

Combining Power Series

Suppose the power series $\sum c_k x^k$ and $\sum d_k x^k$ converge absolutely to $f(x)$ and $g(x)$, respectively, on an interval I.

1. **Sum and difference:** The power series $\sum (c_k \pm d_k) x^k$ converges absolutely to $f(x) \pm g(x)$ on I.

2. **Multiplication by a power:** The power series $x^m \sum c_k x^k = \sum c_k x^{k+m}$ converges absolutely to $x^m f(x)$ on I, provided m is an integer such that $k + m \geq 0$ for all terms of the series.

3. **Composition:** If $h(x) = bx^m$, where m is a positive integer and b is a real number, the power series $\sum c_k(h(x))^k$ converges absolutely to the composite function $f(h(x))$, for all x such that $h(x)$ is in I.

Differentiating and Integrating Power Series

Let the function f be defined by the power series $\sum c_k(x-a)^k$ on its interval of convergence I.

- f is a continuous function on I.
- The power series may be differentiated or integrated term by term, and the resulting power series converges to $f'(x)$ or $\int f(x)\ dx\ +\ C$, respectively, at all points in the interior of I, where C is an arbitrary constant.

10.3 Taylor Series

Taylor/Maclaurin Series for a Function

Suppose the function f has derivatives of all orders on an interval containing the point a. The **Taylor series for f centered at a** is

$$f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^{(3)}(a)}{3!}(x-a)^3 + \cdots \tag{9}$$

$$= \sum_{k=0}^{\infty} \frac{f^{(k)}(a)}{k!}(x-a)^k. \tag{10}$$

A Taylor series centered at 0 is called a **Maclaurin series**.

Binomial Coefficients

$\forall p, k \in \mathbb{R} \land k \geq 1$

$$\binom{p}{k} = \frac{p(p-1)(p-2)\cdots(p-k+1)}{k!}. \tag{11}$$

With the special case of $\binom{p}{0} = 1$.

Binomial Series

$\forall p \in \mathbb{R} \land p \neq 0$, the Taylor series for $f(x) = (1+x)^p$ centered at 0 is the **binomial series**

$$\sum_{k=0}^{\infty} x^k = \sum_{k=0}^{\infty} \frac{p(p-1)(p-2)\cdots(p-k+1)}{k!} x^k \tag{12}$$

$$= 1 + px + \frac{p(p-1)(p-2)}{3!}x^3 + \cdots \tag{13}$$

The series converges for $|x| < 1$ (and possibly at the endpoints, depending on p). If p is a nonnegative integer, the series terminates and results in a polynomial of degree p.

Convergence of Taylor Series

Let f have derivatives of all orders on an open interval I containing a. The Taylor series for f centered at a converges to f , for all x in I, if and only if $\lim_{n\to\infty} R_n(x) = 0$, for all x in I, where

$$R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}(x-a)^{n+1} \tag{14}$$

is the remainder at x (with c between x and a).

Taylor Series Functions

Chapter 11

Course:
../math1215-calculus-ii/chapter-11/chapter-11.pdf

Document 29

Included from: ../math1215-calculus-ii/chapter-11/chapter-11.pdf

00A0

Chapter 11: Parametric and Polar Curves

Illya Starikov

June 30, 2025

Contents

11.1 Parametric Equations

Forward or Positive Orientation

The direction in which a parametric curve is generated as the parameter increases is called the **forward**, or **positive**, **orientation** of the curve.

Derivative for Parametric Curves

Let $x = g(t)$ and $y = h(t)$, where g and h are differentiable on an interval $[a, b]$. Then

$$\frac{dy}{dx} = \frac{dy/dt}{dx/dt} = \frac{y'(t)}{x'(t)} \tag{1}$$

provided $\frac{dx}{dt} \neq 0$.

11.2 Polar Coordinates

Introduction

- Up to now we have only studied in a Cartesian coordinate system.
  - A Cartesian coordinate system is just a plane described by Cartesian (or, algebraic) equations and points in a finite dimensions.
    * *One Dimension*: Lines.
    * *Two Dimensions*: x^2.
    * *Three, Four*: Upper-level Calculus and Physics.
- Let's define an alternative coordinate system — **polar coordinate.**
  - coordinates are constants on circles and rays.
  - Useful for navigation, position, and gravitation fields.

Defining Polar Coordinates

Pole The origin of the coordinate system.

Polar Axis Synonymous for the positive x-axis.

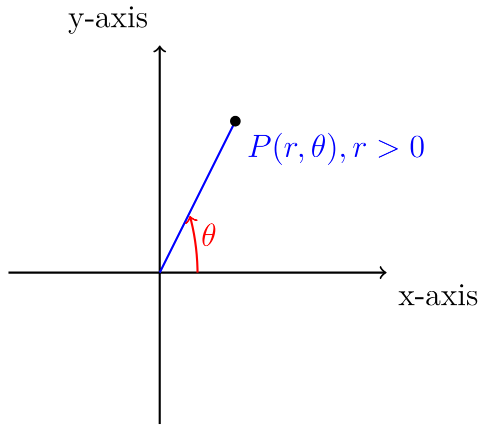
Polar Coordinates A polar coordinates P has the form (r, θ).

> **Radial Coordinate** The radial coordinate r describes the *signed*, or *directed*, distance from the origin to P.
>
> **Angular Coordinate** The angular coordinate θ describes an angle whose initial side is the positive x-axis and whose terminal side lies on the ray passing through the origin and P.

Notes

- **Positive angle measurements are measured *counterclockwise* from the origin.**
- Every point has multiple representations.
  - Angles are periodic, so multiples of 2π gives the same angle.

- Coordinates may be negative. So (r, θ) can be represented as $(-r, \theta + \pi)$ and $(-r, \theta - \pi)$



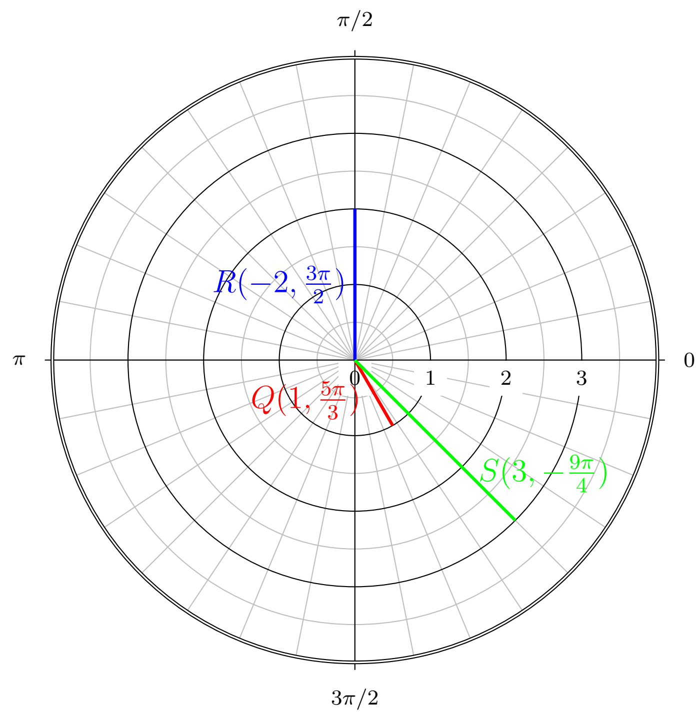
In summary,

- $(r, \theta + 2\pi)$ represents the same point as (r, θ)
- $P(r, \theta)$ and $P'(-r, \theta)$ are reflections through the origin.

Examples

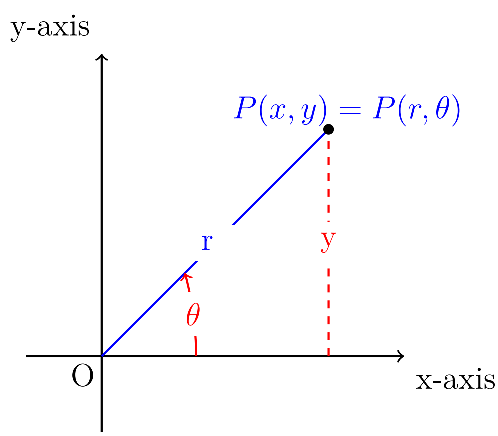
Graph the following points in polar coordinates:

- $Q(1, \frac{5\pi}{3})$
- $R(-2, \frac{3\pi}{2})$
- $S(3, -\frac{9\pi}{4})$
  - Now give two alternative representations.
  - $S'(3, \frac{1\pi}{4})$
  - $S''(-3, -\frac{5\pi}{4})$



Converting Between Cartesian and Polar Coordinates

- We sometimes need to convert between Cartesian and polar coordinates.
- Let's turn this problem into a right triangle.



A point with polar coordinates (r, θ) has Cartesian coordinates (x, y), where

$$x = r\cos\theta \quad \text{and} \quad y = r\sin\theta \tag{2}$$

A point with Cartesian coordinates (x, y) has polar coordinates (r, θ), where

$$r^2 = x^2 + y^2 \quad \text{and} \quad \tan\theta = \frac{y}{x} \tag{3}$$

Examples

BE SURE TO GRAPH POINTS IN CARTESIAN FIRST.

Polar to Cartesian

Express the point with the following polar coordinates in Cartesian coordinates: $P(3, \frac{2\pi}{3})$

$$\begin{aligned} x &= r\cos\theta \\ &= 3\cos(2\pi/3) \\ &= -3(1/2) \\ &= -3/2 \end{aligned} \qquad \begin{aligned} y &= r\sin\theta \\ &= 3\sin(2\pi/3) \\ &= 3(\sqrt{3}/2) \\ &= 3\sqrt{3}/2 \end{aligned}$$

Polar to Cartesian

Express the point with the following polar coordinates in Cartesian coordinates: $Q(e, -\frac{\pi}{4})$

$$\begin{aligned} x &= r\cos\theta \\ &= e\cos(-\pi/4) \\ &= e(\sqrt{2}/2) \\ &= e\sqrt{2}/2 \end{aligned} \qquad \begin{aligned} y &= r\sin\theta \\ &= e\sin(-\pi/4) \\ &= -e(\sqrt{2}/2) \\ &= -e\sqrt{2}/2 \end{aligned}$$

Cartesian To Polar

Express the point with the following Polar coordinates to Cartesian coordinates: $R(1, -1)$

$$\begin{aligned} r &= \sqrt{x^2 + y^2} \\ &= \sqrt{1^2 + (-1)^2} \\ &= \sqrt{2} \end{aligned} \qquad \begin{aligned} \tan\theta &= {}^{y}\!/_{x} \\ &= {}^{-1}\!/_{1} \\ &= -1 \\ \theta &= {}^{-\pi}\!/_{4} \text{ or } {}^{7\pi}\!/_{4}. \end{aligned}$$

Therefore, two possible solutions are: $(\sqrt{2}, {}^{-\pi}\!/_{4})$ or $(\sqrt{2}, {}^{7\pi}\!/_{4})$

Cartesian To Polar

Express the point with the following Polar coordinates to Cartesian coordinates: $S(1, \sqrt{3})$

$$\begin{aligned} r &= \sqrt{x^2 + y^2} \\ &= \sqrt{(\sqrt{3})^2 + (1)^2} \\ &= 2 \end{aligned} \qquad \begin{aligned} \tan\theta &= {}^{y}\!/_{x} \\ &= {}^{\sqrt{(3)}}\!/_{1} \\ \theta &= {}^{\pi}\!/_{3} \text{ or } {}^{4\pi}\!/_{3}. \end{aligned}$$

Therefore, two possible solutions are: $(2, {}^{\pi}\!/_{3})$ or $(2, {}^{4\pi}\!/_{3})$.

Basic Curves in Polar Coordinates

- A curve in polar coordinates is the set of **points** that satisfy an equation in r and θ.
- This makes graphing some things easier than others.
- Look at $r = 3$ is the set of all points that satisfy being away from the origin of 3 units.
  - This is because θ is not specified, it's arbitrary. Basically, θ is the function.

- In general, $r = a$, $\forall a \in \mathbb{R}^+$ describes a circle.

- Taking the converse, let r be arbitrary.
  - If the r is arbitrary, and we specify the angle, what do you think we get?
  - A line!
  - Take $\sqrt{2}/2$.

Polar to Cartesian Graph Example

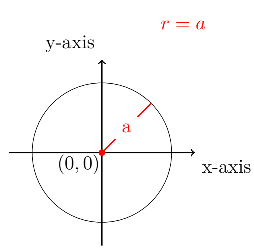
Convert the polar equation $r = 6\sin\theta$ to Cartesian coordinates and describe the corresponding graph.

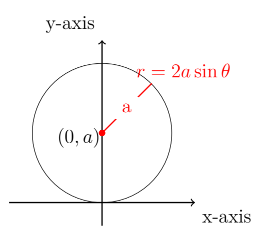
$$\begin{aligned} r^2 &= 6r\sin\theta && (4)\\ x^2+y^2 &= 6y && (5)\\ 0 &= x^2+y^2-6y && (6)\\ &= x^2+(y^2-6y+9)-9 && (7)\\ &= x^2+(y-3)^2-9 && (8) \end{aligned}$$

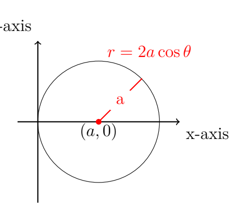
We recognize this to be the equation of a circle, centered at $(0,3)$ at 3. We can also generalize this.

Circle in Polar Coordinates

- The equation $r = a$ describes a circle of radius $|a|$ centered at $(0,0)$.
- The equation $r = 2a\sin\theta$ describes a circle of radius $|a|$ centered at $(0,a)$.
- The equation $r = 2a\cos\theta$ describes a circle of radius $|a|$ centered at $(a,0)$.



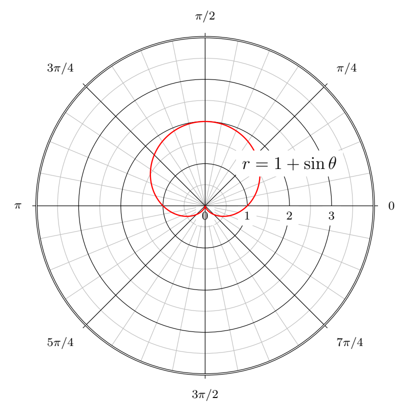




Graphing In Polar Coordinates

Graph the polar equation $r = f(\theta) = 1 + \sin\theta$

θ	$r = 1 + sin\theta$
0	1
$^{\pi}/_{6}$	$^{3}/_{2}$
$^{\pi}/_{2}$	2
$^{5\pi}/_{6}$	$^{3}/_{2}$
π	1
$^{7\pi}/_{6}$	$^{1}/_{2}$
$^{3\pi}/_{2}$	0
$^{11\pi}/_{6}$	$^{1}/_{2}$
2π	1



The resulting curve is known as a **cardioid**.

Cartesian-to-Polar Method for Graphing $r = f(\theta)$

1. Graph $r = f(\theta)$ *as if* r *and* θ *were Cartesian coordinates* with θ on the horizontal axis and r on the vertical axis. Be sure to choose an interval in θ on which the entire polar curve is produced.

2. Use the Cartesian graph in Step 1 as a guide to sketch the points (r, θ) on the final *polar* curve.

Example

With the alternate graphing method, graph $r = 1 + \sin\theta$.

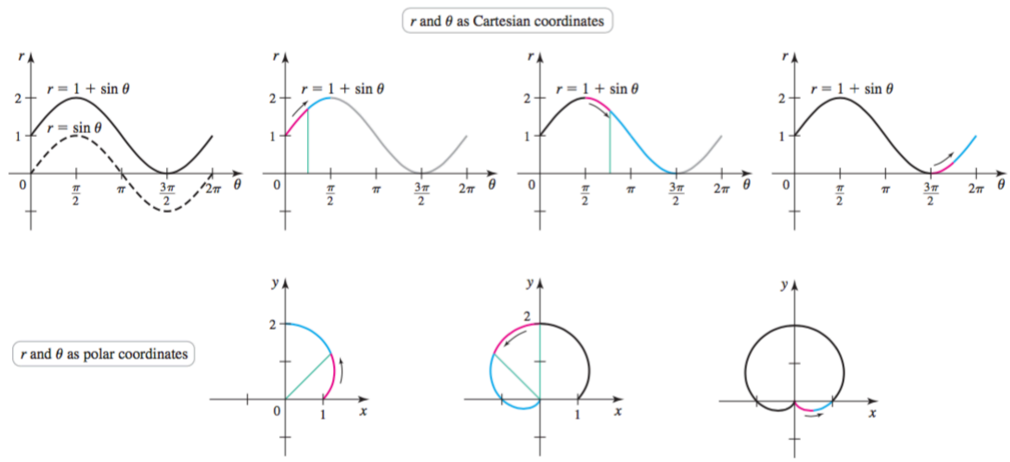


FIGURE 11.26

Symmetry In Polar Equations

Symmetry about the x-axis occurs if the point (r, θ) is on the graph whenever $(r, -\theta)$ is on the graph.

Symmetry about the y-axis occurs if the point (r, θ) is on the graph whenever $(r, \pi - \theta) = (-r, -\theta)$ is on the graph.

Symmetry about the origin occurs if the point $(r, \theta$ is on the graph whenever $(-r, \theta) = (r, \theta + \pi)$ is on the graph.

11.3 Calculus in Polar Coordinates

Slope of a Tangent Line

Let f be a differentiable function at θ_0. The slope of the line tangent to the curve $r = f(\theta)$ at the point $(f(\theta_0, \theta_0)$ is

$$\frac{dy}{dx} = \frac{f'(\theta_0)\sin\theta_0 + f(\theta_0)\cos\theta_0}{f'(\theta_0)\cos\theta_0 - f(\theta_0)} \tag{9}$$

Area of Regions in Polar Coordinates

Let R be the region bounded by the graphs of $r = f(\theta)$ and $r = g(\theta)$, between $\theta = \alpha$ and $\theta = \beta$, where f and g are continous and $f(\theta) \geq g(\theta) \geq 0$ on $[\alpha, \beta]$. The area of R is

$$\int_\alpha^\beta \frac{1}{2}(f(\theta)^2 - g(\theta)^2)\, d\theta \tag{10}$$

Chapter 11

MATH 2100 - Foundations of Mathematics

Homework 1

Course:
../math2100-foundations-of-mathematics/homeworks/homewor

Document 30

Included from:
../math2100-foundations-of-mathematics/homeworks/homework-1.pdf

Homework #1

Illya Starikov

Thursday, January 26^{th}, 2017

Theorem 1. *There does not exist a level set with more than one bottom level.*

Proof. Suppose that there exists two, different levels P and Q in the level set M (where M may contain two — P and Q — or more levels). Per our definition, P and Q are bottom levels of the level set M if there exists a level x in M where x is not above P and Q. This implies P cannot be above Q and Q cannot be above P, for a level x must not be above P and Q. However, this violates Axiom 3 (which states that P is above Q or Q is above P). This has lead us to a contradiction. $\therefore$ There does not exists a level set with more than one bottom level. $\square$

Chapter 12

MATH 2222 - Calculus III

Chapter 12

Course:
`../math2222-calculus-iii/chapter-12/chapter-12.pdf`

Document 31

Included from: `../math2222-calculus-iii/chapter-12/chapter-12.pdf`

Chapter 12: Vectors and Vector Valued Functions

Illya Starikov

June 30, 2025

Contents

12.1 Vectors in the Plane

Vectors, Equal Vectors, Scalars, Zero Vector

Vectors are quantities that have both **length** (or **magnitude**) and **direction**. Two vectors are **equal** if they have the same magnitude and direction. Quantities having magnitude but no direction are called **scalars**. One exception is the **zero** vector, denoted $\mathbf{0}$: It has length 0 and no direction.

Scalar Multiples and Parallel Vectors

Given a scalar c and a vector $\mathbf{u}$, the scalar multiple $c\mathbf{v}$ is a vector whose magnitude is $|c|$ multiplied by the magnitude of $\mathbf{v}$. If $c > 0$, then $c\mathbf{v}$ has the same direction as $\mathbf{v}$. If $c < 0$, then $c\mathbf{v}$ and $\mathbf{v}$ point in opposite directions. Two vectors are **parallel** if they are scalar multiples of each other.

Position Vectors and Vector Components

A vector $\mathbf{v}$ with its tail at the origin and head at the point (v_1, v_2) is called a **position vector** (or is said to be in **standard position**) and is written $\langle v_1, v_2\rangle$. The real numbers v_1 and v_2 are the x- and y-**components** of $\mathbf{v}$, respectively. The position vectors $\mathbf{u} = \langle v_1, v_2\rangle$ and $\mathbf{v} = \langle v_1, v_2\rangle$ are equal if and only if $u_1 = v_1$ and $u_2 = v_2$.

Magnitude of a Vector

Given the points $P(x_1, y_1)$ and $Q(x_2, y_2)$, the **magnitude**, or **length**, of $\vec{PQ} = \langle x_2 - x_1, y_2 - y_1\rangle$, denoted $|\vec{PQ}|$, is the distance between P and Q:

$$|\vec{PQ}| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1}$$

The magnitude of the position vector $\mathbf{v} = \langle v_1, v_2\rangle$ is $|\mathbf{v}| = \sqrt{v_1^2 + v_2^2}$

Vector Operations

Suppose c is a scalar, $\mathbf{u} = \langle u_1, u_2\rangle$ and $\mathbf{v} = \langle v_1, v_2\rangle$.

$$\mathbf{u} + \mathbf{v} = \langle u_1 + v_1, u_2 + v_2\rangle \qquad \text{Vector addition} \tag{2}$$

$$\mathbf{u} - \mathbf{v} = \langle u_1 - v_1, u_2 - v_2\rangle \qquad \text{Vector subtraction} \tag{3}$$

$$c\mathbf{u} = \langle cu_1, cu_2\rangle \qquad \text{Scalar multiplication} \tag{4}$$

Unit Vectors and Vectors of a Specified Length

A **unit vector** is any vector with length 1. Given a nonzero vector $\mathbf{v}, \pm\frac{\mathbf{v}}{|\mathbf{v}|}$ are unit vectors parallel to $\mathbf{v}$. For a scalar $c > 0$, the vectors $\pm\frac{c\mathbf{v}}{|v|}$ are vectors of length c parallel to $\mathbf{v}$.

Properties of Vector Operations

Suppose $\mathbf{u}, \mathbf{v}$, and $\mathbf{w}$ are vectors and a and c are scalars. Then the following properties hold (for vectors in any number of dimensions).

$$\mathbf{u} + a = \mathbf{v} + \mathbf{u} \quad \text{Commutative property of addition} \tag{5}$$

$$(\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w}) \quad \text{Associative property of addition} \tag{6}$$

$$\mathbf{v} + \mathbf{0} = \mathbf{v} \quad \text{Additive identity} \tag{7}$$

$$\mathbf{v} + (-\mathbf{v}) = 0 \quad \text{Additive identity} \tag{8}$$

$$c(\mathbf{u} + \mathbf{v}) = c\mathbf{u} + c\mathbf{v} \quad \text{Distributive property 1} \tag{9}$$

$$(a + c)\mathbf{v} = a\mathbf{v} + c\mathbf{v} \quad \text{Distributive property 2} \tag{10}$$

$$0\mathbf{v} = \mathbf{0} \quad \text{Multiplication by zero scalar} \tag{11}$$

$$c\mathbf{0} = \mathbf{0} \quad \text{Multiplication by zero vector} \tag{12}$$

$$1\mathbf{v} = \mathbf{v} \quad \text{Multiplicative identity} \tag{13}$$

$$a(c\mathbf{v}) = (ac)\mathbf{v} \quad \text{Associative property of scalar multiplication} \tag{14}$$

12.2 Vectors in Three Dimensions

Distance Formula in xyz-Space

The distance between the points $P(x_1, y_1, z_1)$ and $Q(x_2, y_2, z_2)$ is

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{15}$$

Spheres and Balls

A **sphere** centered at (a, b, c) with radius r is the set of points satisfying the equation

$$(x - a)^2 + (y - b)^2 + (z - c)^2 = r^2 \tag{16}$$

A ball centered at (a, b, c) with radius r is the set of points satisfying the inequality

$$(x - a)^2 + (y - b)^2 + (z - c)^2 \leq r^2 \tag{17}$$

Vector Operations in $\mathbb{R}$

Let c be a scalar, $\mathbf{u} = \langle u_1, u_2, u_3 \rangle$ and $\mathbf{v} = \langle v_1, v_2, v_3 \rangle$.

$$\mathbf{u} + \mathbf{v} = \langle u_1 + v_1, u_2 + v_2, u_3 + v_3 \rangle \quad \text{Vector addition} \tag{18}$$

$$\mathbf{u} - \mathbf{v} = \langle u_1 - v_1, u_2 - v_2, u_3 - v_3 \rangle \quad \text{Vector subtraction} \tag{19}$$

$$c\mathbf{u} = \langle cu_1, cu_2, c_3 \rangle \tag{20}$$

Magnitude of a Vector

The **magnitude** (or **length**) of the vector $\vec{PQ} = \langle x_2 - x_1, y_2 - y_1, z_2 - z_1 \rangle$ is the distance from $P(x_1, y_1, z_1)$ to $Q(x_2, y_2, z_2)$

$$|\vec{PQ}| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{21}$$

12.3 Dot Product

Dot Product

Given two nonzero vectors $\mathbf{u}$ and $\mathbf{v}$ in two or three dimensions, their **dot product** is

$$\mathbf{u} \cdot \mathbf{v} = |\mathbf{u}||\mathbf{v}| \cos\theta \tag{22}$$

where θ is the angle between $\mathbf{u}$ and $\mathbf{v}$ with $0 \leq \theta \leq \pi$. If $\mathbf{u} = \mathbf{0}$ or $\mathbf{v} = \mathbf{0}$, then $\mathbf{u} \cdot \mathbf{v} = 0$, and θ is undefined.

Orthogonal Vectors

Two vectors $\mathbf{u}$ and $\mathbf{v}$ are **orthogonal** if and only if $\mathbf{u} \cdot \mathbf{v} = 0$. The zero vector is orthogonal to all vectors. In two or three dimensions, two nonzero orthogonal vectors are perpendicular to each other.

Dot Product

Given two vectors $\mathbf{u} = \langle u_1, u_2, u_3 \rangle$ and $\mathbf{v} = \langle v_1, v_2, v_3 \rangle$,

$$\mathbf{u} \cdot \mathbf{v} = u_1 v_1 + u_2 v_2 + u_3 v_3 \tag{23}$$

Properties of the Dot Product

Suppose $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ are vectors and let c be a scalar.

$$\mathbf{u} \cdot \mathbf{v} = \mathbf{v} \cdot \mathbf{u} \qquad \text{Commutative property} \tag{24}$$

$$c(\mathbf{u} \cdot \mathbf{v}) = (c\mathbf{u}) \cdot \mathbf{v} = \mathbf{u} \cdot (c\mathbf{v}) \qquad \text{Associative property} \tag{25}$$

$$\mathbf{u} \cdot (\mathbf{v} + \mathbf{w}) = \mathbf{u} \cdot \mathbf{v} + \mathbf{u} \cdot \mathbf{w} \tag{26}$$

(Orthogonal) Projection of u onto v

The **orthogonal projection of u onto v** , denoted $\text{proj}_{\mathbf{v}}\mathbf{u}$, where $\mathbf{v} \neq \mathbf{0}$, is

$$\text{proj}_{\mathbf{v}}\mathbf{u} = |\mathbf{u}| \cos\theta \left(\frac{\mathbf{v}}{|\mathbf{v}|} \right) \tag{27}$$

The orthogonal projection may also be computed with the formulas

$$\text{proj}_{\mathbf{v}}\mathbf{u} = \text{scal}_{\mathbf{v}}\mathbf{u}\left(\frac{\mathbf{v}}{|\mathbf{v}|}\right) = \left(\frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{v}\cdot\mathbf{v}}\right)\mathbf{v} \tag{28}$$

where the **scalar component of u in the direction of v** is

$$\text{scal}_{\mathbf{v}}\mathbf{u} = |\mathbf{u}|\cos\theta = \frac{\mathbf{u}\cdot\mathbf{v}}{|\mathbf{v}|} \tag{29}$$

Work

Let a constant force $\mathbf{F}$ be applied to an object, producing a displacement $\mathbf{d}$. If the angle between $\mathbf{F}$ and $\mathbf{d}$ is θ, then the **work** done by the force is

$$W = |\mathbf{F}||\mathbf{d}|\cos\theta = \mathbf{F}\cdot\mathbf{d} \tag{30}$$

12.4 Cross Product

Given two nonzero vectors $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{R}^3$, the **cross product** $\mathbf{u} \times \mathbf{v}$ is a vector with magnitude

$$|\mathbf{u} \times \mathbf{v}| = |\mathbf{u}||\mathbf{v}| \sin\theta \tag{31}$$

where $0 \leq \theta \leq \pi$ is the angle between $\mathbf{u}$ and $\mathbf{v}$. The direction of $\mathbf{u} \times \mathbf{v}$ is given by the **right-hand rule**: When you put the vectors tail to tail and let the fingers of your right hand curl from $\mathbf{u}$ to $\mathbf{v}$ the direction of $\mathbf{u} \times \mathbf{v}$ is the direction of your thumb, orthogonal to both $\mathbf{u}$ and $\mathbf{v}$. When $\mathbf{u} \times \mathbf{v} = 0$, the direction of $\mathbf{u} \times \mathbf{v}$ undefined.

Geometry of the Cross Product

Let $\mathbf{u}$ and $\mathbf{v}$ be two nonzero vectors in $\mathbb{R}^3$.

1. The vectors $\mathbf{u}$ and $\mathbf{v}$ are parallel ($\theta = 0$ or $\theta = \pi$) if and only if $\mathbf{u} \times \mathbf{v} = 0$.

2. If $\mathbf{u}$ and $\mathbf{v}$ are two sides of a parallelogram, then the area of the parallelogram is

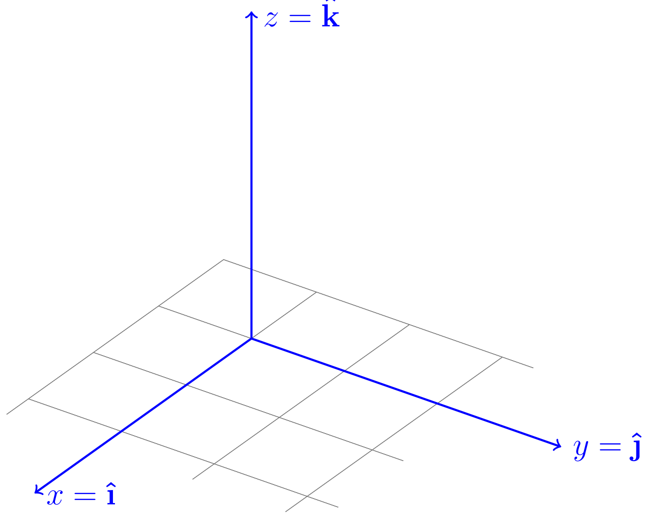
$$|\mathbf{u} \times \mathbf{v}| = |\mathbf{u}||\mathbf{v}| \sin\theta \tag{32}$$

Properties of the Cross Product

Let $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ be nonzero vectors in $\mathbb{R}^3$, and let a and b be scalars.

$$\begin{aligned}
\mathbf{u} \times \mathbf{v} &= -(\mathbf{v} \times \mathbf{u}) && \text{Anticommutative property} && (33)\\
(a\mathbf{u}) \times (b\mathbf{v}) &= ab(\mathbf{u} \times \mathbf{v}) && \text{Associative property} && (34)\\
\mathbf{u} \times (\mathbf{v} + \mathbf{w}) &= (\mathbf{u} \times \mathbf{v}) + (\mathbf{u} \times \mathbf{w}) && \text{Distributive property} && (35)\\
(\mathbf{u} + \mathbf{v}) \times \mathbf{w} &= (\mathbf{u} \times \mathbf{w}) + (\mathbf{v} \times \mathbf{w}) && \text{Distributive property} && (36)
\end{aligned}$$

Cross Products of Coordinate Unit Vectors



$$\hat{\mathbf{i}} \times \hat{\mathbf{j}} = -(\hat{\mathbf{j}} \times \hat{\mathbf{i}}) = \hat{\mathbf{k}} \tag{37}$$

$$\hat{\mathbf{j}} \times \hat{\mathbf{k}} = -(\hat{\mathbf{k}} \times \hat{\mathbf{j}}) = \hat{\mathbf{i}} \tag{38}$$

$$\hat{\mathbf{k}} \times \hat{\mathbf{i}} = -(\hat{\mathbf{i}} \times \hat{\mathbf{k}}) = \hat{\mathbf{j}} \tag{39}$$

$$\hat{\mathbf{i}} \times \hat{\mathbf{i}} = \hat{\mathbf{j}} \times \hat{\mathbf{j}} = \hat{\mathbf{k}} \times \hat{\mathbf{k}} = 0 \tag{40}$$

Evaluating the Cross Product

Let $\mathbf{u} = u_1\hat{\mathbf{i}} + u_2\hat{\mathbf{j}} + u_3\hat{\mathbf{k}}$ and $\mathbf{v} = v_1\hat{\mathbf{i}} + v_2\hat{\mathbf{j}} + v_3\hat{\mathbf{k}}$. Then

$$\mathbf{u} \times \mathbf{v} = \begin{vmatrix} \hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{vmatrix} = \begin{vmatrix} u_2 & u_3 \\ v_2 & v_3 \end{vmatrix} \hat{\mathbf{i}} - \begin{vmatrix} u_1 & u_3 \\ v_1 & v_3 \end{vmatrix} \hat{\mathbf{j}} + \begin{vmatrix} u_1 & u_2 \\ v_1 & v_2 \end{vmatrix} \hat{\mathbf{k}} \tag{41}$$

12.5 Lines and Curves in Space

Equation of a Line

An equation of the line passing through the point $P_0(x_0,\, y_0,\, z_0)$ in the direction of the vector $\mathbf{v} = \langle a,\, b,\, c\rangle$ is $\mathbf{r} = r_0 + t\mathbf{v}$, or

$$\langle x,\, y,\, z\rangle = \langle x_0,\, y_0,\, z_0\rangle + t\langle a,\, b,\, c\rangle, \quad \text{for} \quad -\infty < t < \infty \tag{42}$$

Equivalently, the parametric equations of the line are

$$x = x_0 + at, \quad y = y_0 + bt, \quad z = z_0 + ct, \quad \text{for} \quad -\infty < t < \infty \tag{43}$$

Limit of a Vector-Valued Function

A vector-valued function $\mathbf{r}$ approaches the limit $\mathbf{L}$ as t approaches a, written $\lim_{t \to a} \mathbf{r}(t) = \mathbf{L}$, provided $\lim_{t \to a} |\mathbf{r}(t) - \mathbf{L}| = 0$

12.6 Calculus of Vector-Valued Functions

Derivative and Tangent Vector

Let $\mathbf{r}(t) = f(t)\hat{\mathbf{i}} + g(t)\hat{\mathbf{j}} + h(t)\hat{\mathbf{k}}$, where f, g, and h are differentiable functions on $(a\,b)$. Then $\mathbf{r}$ has a **derivative** (or is **differentiable**) on $(a\,b)$ and

$$\mathbf{r}'(t) = f'(i)\hat{\mathbf{i}} + g'(t)\hat{\mathbf{j}} + h'(t)\hat{\mathbf{k}} \tag{44}$$

Provided $\mathbf{r}'(t) \neq \mathbf{0}, \mathbf{r}'(t)$ is a **tangent vector** (or velocity vector) at the point corresponding to $\mathbf{r}$.

Unit Tangent Vector

Let $\mathbf{r} = f(t)\hat{\mathbf{i}} + g(t)\hat{\mathbf{j}} + h(t)\hat{\mathbf{k}}$ be a smooth parameterized curve, for $a \leq t \leq b$. The **unit tangent vector** for a particular value of t is

$$\mathbf{T}(t) = \frac{\mathbf{r}'(t)}{|\mathbf{r}'(t)|} \tag{45}$$

Derivative Rules

Let $\mathbf{u}$ and $\mathbf{v}$ be differentiable vector-valued functions and let f be a differentiable scalar-valued function, all at a point t. Let $\mathbf{c}$ be a constant vector. The following rules apply.

$$\frac{d}{dt}(\mathbf{c}) = \mathbf{0} \quad \text{Constant Rule} \tag{46}$$

$$\frac{d}{dt}(\mathbf{u}(t) + \mathbf{v}(t)) = \mathbf{u}'(t) + \mathbf{v}'(t) \quad \text{Sum Rule} \tag{47}$$

$$\frac{d}{dt}(f(t)\mathbf{u}(t)) = f'(t)\mathbf{u}(t) + f(t)\mathbf{u}'(t) \quad \text{Product Rule} \tag{48}$$

$$\frac{d}{dt}(\mathbf{u}(f(t))) = \mathbf{u}'(f(t))f'(t) \quad \text{Chain Rule} \tag{49}$$

$$\frac{d}{dt}(\mathbf{u}(t) \cdot \mathbf{v}(t)) = \mathbf{u}'(t) \cdot \mathbf{v}(t) + \mathbf{u}(t) \cdot \mathbf{v}'(t) \quad \text{Dot Product Rule} \tag{50}$$

$$\frac{d}{dt}(\mathbf{u}(t) \times \mathbf{v}(t)) = \mathbf{u}'(t) \times \mathbf{v}(t) + \mathbf{u}(t) \times \mathbf{v}'(t) \quad \text{Cross Product Rule} \tag{51}$$

Indefinite Integral of a Vector-Valued Function

Let $\mathbf{r} = f\hat{\mathbf{i}} + g\hat{\mathbf{j}} + h\hat{\mathbf{k}}$ be a vector function and let $\mathbf{R} = F\hat{\mathbf{i}} + G\hat{\mathbf{j}} + H\hat{\mathbf{k}}$, where F, G, and H are antiderivatives of f, g, and h, respectively. The **indefinite integral** of $\mathbf{r}$ is

$$\int \mathbf{r}(t)\, dt = \mathbf{R}(t) + \mathbf{C} \tag{52}$$

where $\mathbf{C}$ is an arbitrary constant vector.

Definite Integral of a Vector-Valued Function

Let $\mathbf{r}(t) = f(t)\hat{\mathbf{i}} + g(t)\hat{\mathbf{j}} + h(t)\hat{\mathbf{k}}$, where f, g, and h are integrable on the interval $[a, b]$.

$$\int \mathbf{r}(t)\, dt = \left[\int_a^b f(t)\, dt\right]\hat{\mathbf{i}} + \left[\int_a^b g(t)\, dt\right]\hat{\mathbf{j}} + \left[\int_a^b h(t)\, dt\right]\hat{\mathbf{k}} \tag{53}$$

12.7 Motion In Space

Position, Velocity, Speed, Acceleration

Let the position of an object moving in three-dimensional space be given by $\mathbf{r}(t) = \langle x(t), y(t), z(t) \rangle$, for $t \geq 0$. The **velocity** of the object is

$$\mathbf{v}(t) = \mathbf{r}'(t) = \langle x'(t), y'(t), z'(t) \rangle \tag{54}$$

The **speed** of the object is the scalar function

$$|\mathbf{v}(t)| = \sqrt{x'(t)^2 + y'(t)^2 + z'(t)^2} \tag{55}$$

The **acceleration** of the object is $\mathbf{a}(t) = \mathbf{v}'(t) = \mathbf{r}''(t)$.

Motion with Constant $|\mathbf{r}|$

Let $\mathbf{r}$ describe a path on which $|\mathbf{r}|$ is constant (motion on a circle or sphere centered at the origin). Then, $\mathbf{r} \cdot \mathbf{v} = 0$, which means the position vector and the velocity vector are orthogonal at all times for which the functions are defined.

Two-Dimensional Motion in a Gravitational Field

Consider an object moving in a plane with horizontal x-axis and a vertical y-axis, subject only to the force of gravity. Given the initial velocity $\mathbf{v}(0) = \langle u_0,\, v_0 \rangle$ and the initial position $\mathbf{r}(0) = \langle x_0,\, y_0 \rangle$, the velocity of the object, for $t \geq 0$, is

$$\mathbf{v}(t) = \langle x'(t),\, y'(t) \rangle = \langle u_0,\, -gt + v_0 \rangle \tag{56}$$

and the position is

$$\mathbf{r}(t) = \langle x(t),\, y(t) \rangle = \left\langle u_0\, t + x_0,\, -\frac{1}{2}gt^2 + v_0 t + y_0 \right\rangle \tag{57}$$

Two-Dimensional Motion

Assume an object traveling over horizontal ground, acted on only by the gravitational force, has an initial position $\langle x_0,\, y_0 \rangle = \langle 0,\, 0 \rangle$ and initial velocity

$\langle u_0,\, v_0 \rangle = \langle |\mathbf{v}_0| \cos\alpha,\, |\mathbf{v}_0| \sin\alpha \rangle$. The trajectory, which is a segment or a parabola, has the following properties.

$$\text{time of flight} = T = \frac{2|\mathbf{v}_0| \sin\alpha}{g} \tag{58}$$

$$\text{range} = \frac{|\mathbf{v}_0| \sin 2\alpha}{g} \tag{59}$$

$$\text{maximum height} = y\left(\frac{T}{2}\right) = \frac{(|\mathbf{v}_0| \sin\alpha)^2}{2g} \tag{60}$$

12.8 Length of Curves

Consider the parameterized curve $\mathbf{r}(t) = \langle f(t), g(t), h(t) \rangle$, where f', g', and h' are continuous, and the curve is traversed once for $a \leq t \leq b$. The **arc length** of the curve between $(f(a), g(a), h(a))$ and $(f(b), g(b), h(b))$ is

$$L = \int_a^b \sqrt{f'(t)^2 + g'(t)^2 + h'(t)^2}\, dt = \int_a^b |\mathbf{r}'(\mathbf{t})|\, dt \tag{61}$$

Arc Length of a Polar Curve

Let f have a continuous derivative on the interval $[\alpha, \beta]$. The **arc length** of the polar curve $r = f(\theta)$ on $[\alpha,\ \beta]$ is

$$L = \int_\alpha^\beta \sqrt{f(\theta)^2 + f'(\theta)^2}\, d\theta. \tag{62}$$

Arc Length as a Function of a Parameter

Let $\mathbf{r}$(t) describe a smooth curve, for $t \geq a$. The arc length is given by

$$s(t) = \int_a^t |\mathbf{v}(u)|\, du, \tag{63}$$

where $|\mathbf{v}| = |\mathbf{r}'|$. Equivalently, $\frac{ds}{dt} = |\mathbf{v}t| > 0$. If $|\mathbf{v}(t)| = 1$, for all $t \geq a$, then the parameter t corresponds to arc length.

Arc Length as a Function of a Parameter

Let $\mathbf{r}$(t) describe a smooth curve, for $t \geq a$. The arc length is given by

$$s(t) = \int_a^t |\mathbf{v}(u)|\, du, \tag{64}$$

where $|\mathbf{v}| = |\mathbf{r}'|$. Equivalently, $\frac{ds}{dt} = \mathbf{v}(t) > 0$. If $|\mathbf{v}(\mathbf{t})| = 1$, for all $t \geq a$, then the parameter t corresponds to arc length.

12.9 Curvature and Normal Vectors

Curvature

Let $\mathbf{r}$ describe a smooth parameterized curve. If s denotes arc length and $\mathbf{T} = \frac{\mathbf{r}'}{|\mathbf{r}'|}$ is the unit tangent vector, the **curvature** is $\kappa(s) = |\frac{d\mathbf{T}}{ds}|$

Curvature Formula

Let $\mathbf{r}(t)$ describes a smooth parameterized curve, where t is any parameter. If $\mathbf{v} = \mathbf{r}'$ is the velocity and $\mathbf{T}$ is the unit tangent vector, then the curvature is

$$\kappa(t) = \frac{1}{|\mathbf{v}|}\left|\frac{d\mathbf{T}}{dt}\right| = \left|\frac{\mathbf{T}'(t)}{\mathbf{r}'(t)}\right| \tag{65}$$

Alternative Curvature Formula

Let $\mathbf{r}$ be the position of an object moving on a smooth curve. The **curvature** at a point on the curve is

$$\kappa = \frac{|\mathbf{v} \times \mathbf{a}|}{|\mathbf{v}|^3}, \tag{66}$$

where $\mathbf{v} = \mathbf{r}'$ is the velocity and $\mathbf{a} = \mathbf{v}'$ is the acceleration.

Principal Unit Normal Vector

Let $\mathbf{r}$ describe a smooth parameterized curve. The **principal unit normal vector** at a point P on the curve at which $\kappa \neq 0$ is

$$\mathbf{N}(s) = \frac{d\mathbf{T}/ds}{|d\mathbf{T}/ds|} = \frac{1}{\kappa}\frac{d\mathbf{T}}{ds} \tag{67}$$

In practice, we use the equivalent formula

$$\mathbf{N}(t) = \frac{d\mathbf{T}/dt}{|d\mathbf{T}/dt|} \tag{68}$$

evaluated at the value of t corresponding to P.

Properties of the Principal Unit Normal Vector

Let $\mathbf{r}$ describe a smooth parameterized curve with unit tangent vector $\mathbf{T}$ and principal unit normal vector $\mathbf{N}$.

1. $\mathbf{T}$ and $\mathbf{N}$ are orthogonal at all points of the curve; that is, $\mathbf{T}(t)\cdot\mathbf{N}(t) = 0$, at all points where $\mathbf{N}$ is defined.

2. The principal unit normal vector points to the inside of the curve—in the direction that the curve is turning.

Tangential and Normal Components of the Acceleration

The acceleration vector of an object moving in space along a smooth curve has the following representation in terms of its **tangential component** a_T (in the direction of $\mathbf{T}$) and its normal component a_N (in the direction of $\mathbf{N}$):

$$\mathbf{a} = a_N\mathbf{N} + a_T\mathbf{T}, \tag{69}$$

where $a_N = \kappa|\mathbf{v}|^2 = \frac{|\mathbf{v}\times\mathbf{a}|}{|\mathbf{v}|}$ and $a_T = \frac{d^2s}{dt^2}$.

Unit Binormal Vector and Torsion

Let C be a smooth parameterized curve with unit tangent and principal unit normal vectors $\mathbf{T}$ and $\mathbf{N}$, respectively. Then, at each point of the curve at which the curvature is nonzero, the **unit binormal vector** is

$$\mathbf{B} = \mathbf{T}\times\mathbf{N} \tag{70}$$

and the **torsion** is

$$\tau = -\frac{d\mathbf{B}}{ds}\cdot\mathbf{N} \tag{71}$$

Formulas for Curves in Space

1. Position function: $\mathbf{r}(t) = \langle x(t),\, y(t),\, z(t)\rangle$

2. Velocity: $\mathbf{v} = \mathbf{r}'$

3. Acceleration: $\mathbf{a} = \mathbf{v}'$

4. Unit tangent vector: $\mathbf{T} = \frac{\mathbf{v}}{|\mathbf{v}|}$

5. Principal unit normal vector: $\mathbf{N} = \frac{d\mathbf{T}/dt}{|d\mathbf{T}/dt|}$ (provided $d\mathbf{T}/dt \neq \mathbf{0}$)

6. Curvature: $\kappa = \frac{d\mathbf{T}}{ds} = \frac{1}{|\mathbf{v}|}\left|\frac{d\mathbf{T}}{dt}\right| = \left|\frac{\mathbf{T}'(t)}{\mathbf{r}'(t)}\right| = \frac{|\mathbf{v}\times\mathbf{a}|}{|\mathbf{v}|^3}$

7. Components of acceleration: $\mathbf{a} = a_N\mathbf{N} + a_T\mathbf{T}$, where $a_N = \kappa|\mathbf{v}|^2 = \frac{|\mathbf{v}\times\mathbf{a}|}{|\mathbf{v}|}$ and $a_T = \frac{d^2s}{dt^2} = \frac{\mathbf{v}\cdot\mathbf{a}}{|\mathbf{v}|}$

8. Unit binormal vector: $\mathbf{B} = \mathbf{B} \times \mathbf{N} = \frac{\mathbf{v}\times\mathbf{a}}{|\mathbf{v}\times\mathbf{a}|}$

9. Torsion $\tau = -\frac{d\mathbf{B}}{ds} \cdot \mathbf{N} = \frac{(\mathbf{v}\times\mathbf{a})\cdot\mathbf{a}'}{|\mathbf{v}\times\mathbf{a}|^2} = \frac{(\mathbf{r}'\times\mathbf{r}'')\cdot\mathbf{r}'''}{(\mathbf{r}'\times\mathbf{r}'')^2}$

Chapter 14

Course:
`../math2222-calculus-iii/chapter-14/chapter-14.pdf`

Document 32

Included from: `../math2222-calculus-iii/chapter-14/chapter-14.pdf`

Chapter 14: Multiple Integration

Illya Starikov

June 30, 2025

Contents

14.1 Double Integrals over Rectangular Regions

Volumes and Double Integrals

A function f defined on a rectangular region R in the xy-plane is **integratabtle** on R if $\lim\limits_{\Delta\to 0} \sum_{k=1}^{n} f(x_k^*, y_k^*)\Delta A_k$ exists fora ll partitions of R and for all choices of (x_k^*, y_k^*) within those partitions. The limit is the **double integral of f over** R, which we write

$$\iint\limits_R f(x,\, y)\, dA = \lim_{\Delta\to 0} \sum_{k=1}^{n} f(x_k^*,\, y_k^*)\Delta A_k \tag{1}$$

If f is nonnegative on R, then the double integral equals the volume of the solid bounded by $z = f(x,\, y)$ and the xy-plane over R.

Double Integrals on Rectangular Regions

Let f be continuous on the rectangular region $R = \{(x,\, y) : a \le x \le b,\ c \le y \le d\}$. The double integral of f over R may be evaluated by either of two iterated integrals:

$$\iint\limits_R f(x,\, y)\, dA = \int_c^d \int_a^b dx\, dy = \int_a^b \int_c^d dy\, dx \tag{2}$$

Average Value of a Function over a Plane Region

The **average value** of an integrable function f over a region R is

$$\bar{f} = \frac{1}{\text{area of } R} \iint\limits_R f(x,\, y)\, dA \tag{3}$$

14.2 Double Integrals over General Regions

Let R be a region bounded below and above by the graphs of the continuous functions $y = g(x)$ and $y = h(x)$, respectively, and by the lines $x = a$ and $x = b$. If f is continuous on R, then

$$\iint_R f(x, y)\, dA = \int_a^b \int_{g(x)}^{h(x)} f(x, y)\, dy\, dx \tag{4}$$

Let R be a region bounded on the left and right by the graphs of the continuous functions $x = g(y)$ and $x = h(y)$, respectively, and the lines $y = c$ and $y = d$. If f is continuous on R, then

$$\iint_R f(x,\, y)\, dA = \int_c^d \int_{g(y)}^{h(y)} f(x,\, y)\, dx\, dy \tag{5}$$

Areas of Regions by Double Integrals

Let R be a region in the xy-plane. Then

$$\text{area of } R = \iint_R \tag{6}$$

14.3 Double Integrals in Polar Coordinates

Double Integrals over Polar Rectangular Region

Let f be continuous on the region in the xy-plane $R = \{(r, \theta) : 0 \le a \le r \le b, \alpha \le \theta \le \beta\}$, where $\beta - \alpha \le 2\pi$. Then

$$\iint\limits_R f(r, \theta)\, dA = \int_\alpha^\beta \int_a^b f(r, \theta)\; r\, dr\, d\theta \tag{7}$$

Double Integrals over More General Polar Regions

Let f be continuous on the region in the xy-plane

$$R = \{(r, \theta) : 0 \le g(\theta) \le r \le h(\theta), \alpha \le \theta \le \beta\} \tag{8}$$

where $0 < \beta - \alpha \le 2\pi$. Then

$$\iint\limits_R f(r, \theta)\, dA = \int_\alpha^\beta \int_{g(\theta)}^{h(\theta)} f(r, \theta) r\, dr\, d\theta. \tag{9}$$

Area of Polar Regions

The area of the region $R = \{(r, \theta) : 0 \le g(\theta) \le r \le h(\theta), \alpha \le \theta \le \beta\}$, where $0 < \beta - \alpha \le 2\pi$, is

$$A = \iint\limits_R dA = \int_\alpha^\beta \int_{g(\theta)}^{h(\theta)} r\, dr\, d\theta \tag{10}$$

14.4 Triple Integrals

Let f be continuous over the region

$$D = \{(x,\, y,\, z) : a \le x \le b, g(x) \le y \le h(x),\, G(x,\, y) \le z \le H(x,\, y)\} \tag{11}$$

where g, h, G, and H are continuous functions. Then f is integrable over D and the triple integral is evaluated as the iterated integral

$$\iiint\limits_D f(x,\, y,\, z)\, dV = \int_a^b \int_{g(x)}^{h(x)} \int_{G(x,y)}^{H(x,y)} f(x,\, y,\, z)\, dz\, dy\, dx. \tag{12}$$

Average Value of a Function of Three Variables

If f is continuous on a region D of $\mathbb{R}^3$, then the average value of over D is

$$\bar{f} = \frac{1}{\text{volume } (D)} \iiint\limits_D f(x,\, y,\, z)\, dV \tag{13}$$

14.5 Triple Integrals in Cylindrical and Spherical Coordinates

Transformations Between Cylindrical and Rectangular Coordinates

Rectangular → Cylindrical	**Cylindrical → Rectangular**
$r^2 = x^2 + y^2$	$x = r\cos\theta$
$\tan\theta = {}^{y}\!/_{x}$	$y = r\sin\theta$
$z = z$	$z = z$

Triple Integrals in Cylindrical Coordinates

Let f be continuous over the region

$$D = \{(r,\, \theta,\, z) : g(\theta) \leq r \leq h(\theta),\, \alpha \leq \theta \leq \beta,\, G(x,\, y) \leq z \leq H(x,\, y).\} \quad (14)$$

Then f is integrable over D and the triple integral of f over D in cylindrical coordinates is

$$\iiint_D f(r,\, \theta,\, z)\, dV = \int_\alpha^\beta \int_{g(\theta)}^{h(\theta)} \int_{G(r\cos\theta,\, r\sin\theta)}^{H(r\cos\theta,\, r\sin\theta)} f(r,\, \theta,\, z)\, dz\; r\, dr\, d\theta \quad (15)$$

Tranformations Between Spherical and Rectangular Coordinates

Rectangular → Spherical	**Spherical → Rectangular**
$\rho^2 = x^2 + y^2 + z^2$	$x = \rho\sin\varphi\cos\theta$
Use trigonometry to find φ and θ	$y = \rho\sin\varphi\sin\theta$
	$z = \rho\cos\varphi$

Triple Integrals in Spherical Coordinates

Let f be continuous over the region

$$D = \{(\rho,\, \varphi,\, \theta) : g(\varphi,\, \theta) \leq \rho \leq h(\varphi,\, \theta),\, a \leq \varphi b,\, \alpha \leq \theta \leq \beta\} \quad (16)$$

Then f is integrable over D, and the triple integral of f over D in spherical coordinates is

$$\iiint\limits_D f(\rho,\, \varphi,\, \theta) dV = \int_{\alpha}^{\beta} \int_a^b \int_{g(\varphi,\, \theta)}^{h(\varphi,\, \theta)} f(\rho,\, \varphi,\, \theta) \rho^2 \sin\varphi \; d\rho \, d\varphi \, d\theta \tag{17}$$

14.6 Integrals for Mass Calculations

Center of Mass in One Dimension

Let ρ be an integrable density function on the interval $[a, b]$ (which represents a thin rod or wire). The **center of mass** is location at the point $\bar{x} = \frac{M}{m}$, where the **total moment** M and mass m are

$$M = \int_a^b x\rho(x)\, dx \quad \text{and} \quad m = \int_a^b \rho(x)\, dx \tag{18}$$

Center of Mass in Two Dimensions

Let ρ be integrable density function defined over a closed bounded region R in $\mathbb{R}^2$. The coordinates of the center of mass of the object represented by R are

$$\bar{x} = \frac{M_y}{m} = \frac{1}{m} \iint\limits_R x\rho(x,\, y)\, dA \quad \text{and} \quad \bar{y} = \frac{M_x}{m} = \frac{1}{m} = \iint\limits_R y\rho(x,\, y)\, dA \tag{19}$$

Center of Mass in Three Dimensions

Let ρ be integrable density function on a closed bounded region D in $\mathbb{R}^3$. The coordinates of the center of mass of the region are

$$\bar{x} = \frac{M_{yz}}{m} = \frac{1}{m} \iiint\limits_R x\rho(x,\, y,\, z)\, dV \tag{20}$$

$$\bar{y} = \frac{M_{xz}}{m} = \frac{1}{m} \iiint\limits_R y\rho(x,\, y,\, z)\, dV \tag{21}$$

$$\bar{z} = \frac{M_{xy}}{m} = \frac{1}{m} \iiint\limits_R z\rho(x,\, y,\, z)\, dV \tag{22}$$

where $m = \iiint_D \rho(x,\, y,\, z)\, dV$ is the mass, and M_{yz}, M_{xz} and M_{xy} are the moments with respect to the coordinate planes.

Chapter 15

Course:
`../math2222-calculus-iii/chapter-15/chapter-15.pdf`

Document 33

Included from: `../math2222-calculus-iii/chapter-15/chapter-15.pdf`

Chapter 15: Vector Calculus

Illya Starikov

June 30, 2025

Contents

15.1 Vector Fields

Vector Fields in Two Dimensions

Let f and g be defined on a region R of $\mathbb{R}^2$. A **vector field** in $\mathbb{R}^2$ is a function $\mathbf{F}$ that assigns to each point in R a vector $\langle f(x,y),\, g(x,y)\rangle$. The vector field is written as

$$\mathbf{F}(x,y) = \langle f(x,y),\, g(x,y)\rangle \quad \text{or} \quad \mathbf{F}(x,y) = f(x,y)\hat{\mathbf{\imath}} + g(x,y)\hat{\mathbf{\jmath}} \tag{1}$$

A vector field $\mathbf{F} = \langle f, g\rangle$ is continuous or differentiable on a region R of $\mathbb{R}^2$ if f and g are continuous or differentiable on R, respectively.

Radial Vector Fields in $\mathbb{R}^2$

Let $\mathbf{r} = \langle x,\, y\rangle$. A vector field of the form $\mathbf{F} = f(x,\, y)\mathbf{r}$, where f is a scalar-valued function, is a **radial vector field**. Of specific interest are the radial vector field

$$\mathbf{F}(x,\, y) = \frac{\mathbf{r}}{|\mathbf{r}|^p} = \frac{\langle x,\, y\rangle}{|\mathbf{r}|^p} \tag{2}$$

where p is a real number. At every point (except the origin), the vectors of this field are directed outward from the origin with the magnitude of $|\mathbf{F}| = \frac{1}{|\mathbf{r}|^{p-1}}$.

Vector Fields in Three Dimensions

Let $f,\ g,$ and h be defined on a region D of $\mathbb{R}^3$. A **vector field** in $\mathbb{R}^3$ is a function $\mathbf{F}$ that assigns to each point in D a vector $\langle f(x,\, y,\, z),\, g(x,\, y,\, z),\, h(x,\, y,\, z)\rangle$. The vector field is written as

$$\mathbf{F}(x,\, y,\, z) = \langle f(x,\, y,\, z),\, g(x,\, y,\, z),\, h(x,\, y,\, z)\rangle \quad \text{or} \tag{3}$$

$$\mathbf{F}(x,\, y,\, z) = f(x,\, y,\, z)\hat{\mathbf{\imath}} + g(x,\, y,\, z)\hat{\mathbf{\jmath}} + h(x,\, y,\, z)\hat{\mathbf{k}} \tag{4}$$

A vector field $\mathbf{F} = \langle f,\, g,\, h\rangle$ is continuous or differentiable on a region D of $\mathbb{R}^3$ if $f,\ g,\ h$ are continuous or differentiable on R, respectively. Of particular importance are the **radial vector fields**

$$\mathbf{F}(x,\, y,\, z) = \frac{\mathbf{r}}{|\mathbf{r}|^p} = \frac{\langle x,\, y,\, z\rangle}{|\mathbf{r}|^p} \tag{5}$$

where p is a real number.

Gradient Fields and Potential Functions

Let $z = \varphi(x, y)$ and $w = \varphi(x, y, z)$ be differentiable functions on regions of $\mathbb{R}^2$ and $\mathbb{R}^3$, respectively. The vector field $\mathbf{F} = \nabla\varphi$ is **gradient field**, and the function φ is a **potential function** for $\mathbf{F}$.

15.2 Line Integrals

Scalar Line Integral in the Plane, Arc Length Parameter

Suppose the scalar-valued function f is defined on the smooth curve C : $\mathbf{r}(s) = \langle x(s),\, y(s)\rangle$, parameterized by the arc length s. The **line integral of** f **over** C is

$$\int_C f(x(s),\, y(s))\, ds = \lim_{\Delta \to 0} \sum_{k=1}^{n} f(x(s_k^*),\, y(s_k^*))\Delta s_k, \tag{6}$$

provided this limit exists over all partitions of C. When the limit exists, f is said to be **integrable** on C.

Evaluating Scalar Line Integrals in $\mathbb{R}^2$

Let f be continuous on a region containing a smooth curve C : $\mathbf{r}(t) = \langle x(t),\, y(t)\rangle$, for $a \leq t \leq b$. Then

$$\begin{aligned} \int_C f\, ds &= \int_a^b f(x(t),\, y(t))|\mathbf{r}'(t)|\, dt & (7)\\ &= \int_a^b f(x(t),\, y(t))\sqrt{x'(t)^2 + y'(t)^2}\, dt & (8) \end{aligned}$$

Evaluating the Line Integral $\int_C f\, ds$

1. Find a parametric description of C in the form $\mathbf{r}(t) = \langle x(t),\, y(t)\rangle$, for $a \leq t \leq b$
2. Computer $|\mathbf{r}'(t)| = \sqrt{x'(t)^2 + y'(t)^2}$
3. Make substitutions for x and y in the integrand and evaluate an ordinary integral

$$\int_C f\, ds = \int_a^b f(x(t),\, y(t))|\mathbf{r}'(t)|\, dt \tag{9}$$

Evaluating Scalar Line Integrals in $\mathbb{R}^3$

Let f be continuous on a region containing a smooth curve $C : \mathbf{r}(t) = \langle x(t),\, y(t),\, z(t)\rangle$, for $a \le t \le b$. Then

$$\int f\, ds = \int_a^b f(x(t),\, y(t),\, z(t))|\mathbf{r}'(t)|\, dt \tag{10}$$

$$= \int_a^b f(x(t),\, y(t),\, z(t))\sqrt{x'(t)^2 + y'(t)^2 + z'(t)^2}\, dt \tag{11}$$

Line Integral of a Vector Field

Let $\mathbf{F}$ be a vector field that is continuous on a region containing a smooth oriented curve C parameterized by arc length. Let $\mathbf{T}$ be the unit tangent vector at each point of C consistent with the orientation. The line integral of $\mathbf{F}$ over C is $\int_C \mathbf{F} \cdot \mathbf{T}\, ds$.

Different Forms of Line Integrals of Vector Fields

The line integral $\int_C \mathbf{F} \cdot \mathbf{T}\, ds$ may be expressed in the following forms, where $\mathbf{F} = \langle f,\, g,\, h\rangle$, for $a \le t \le b$:

$$\int_a^b \mathbf{F} \cdot \mathbf{r}'(t)\, dt = \int_a^b (f\, x'(t),\, g\, y'(t),\, h\, z'(t))\, dt \tag{12}$$

$$= \int_C f\, dx + g\, dy + h\, dz \tag{13}$$

$$= \int_C \mathbf{F} \cdot d\mathbf{r} \tag{14}$$

For line integrals in the plane, we let $\mathbf{F} = \langle f,\, g\rangle$ and assume C is parameterized in the form $\mathbf{r}(t) = \langle x(t),\, y(t)\rangle$, for $a \le t \le b$. Then

$$\int_C \mathbf{F} \cdot \mathbf{T}\, ds = \int_a^b (f\, x'(t) + g\, y'(t))\, dt = \int_C f\, dx + g\, dy = \int_C \mathbf{F} \cdot d\mathbf{r} \tag{15}$$

Work Done in a Force Field

Let $\mathbf{F}$ be a continuous force field in a region D of $\mathbb{R}^3$ and let $C : \mathbf{r}(t) = \langle x(t),\, y(t),\, z(t)\rangle$, for $a \le t \le b$, be a smooth curve in D with a unit tangent vector $\mathbf{T}$ consistent with the orientation. The work done in moving an object C in the positive direction is

$$W = \int\limits_C \mathbf{F} \cdot \mathbf{T}\, ds = \int_a^b \mathbf{F} \cdot \mathbf{r}'(\mathbf{t})\, dt \tag{16}$$

Circulation

Let $\mathbf{F}$ be a continuous vector field on a region D of $\mathbb{R}^3$ and let C be a closed smooth oriented curve in D. The **circulation** of $\mathbf{F}$ on C is $\int_C \mathbf{F} \cdot \mathbf{T}\, ds$, where $\mathbf{T}$ is the unit vector tangent to C consistent with the orientation.

Flux

Let $F = \langle f, g\rangle$ be continuous vector field on a region R of $\mathbb{R}^2$. Let $C : \mathbf{r}(t) = \langle x(t),\, y(t)\rangle$, for $a \le t \le b$, be a smooth oriented curve in R that does not intersect itself. The **flux** of the vector field across C is

$$\int\limits_C \mathbf{F} \cdot \mathbf{n}\, ds = \int_a^b (f\, y'(t) - g\, x'(t))\, dt, \tag{17}$$

where $n = \mathbf{T} \times \hat{\mathbf{k}}$ is the unit normal vector and $\mathbf{T}$ is the unit tangent vector consistent with the orientation. If C is a closed curve with counterclockwise orientation, $\mathbf{n}$ is the outward normal vector and the flux integral gives the **outward flux** across C.

15.3 Conservative Vector Fields

Simple and Closed Curves

Suppose a curve C (in $\mathbb{R}^2$ and $\mathbb{R}^3$) is described parametrically by $\mathbf{r}(t)$, where $a \le t \le b$. Then C is a **simple curve** if $\mathbf{r}(t_1) \neq \mathbf{r}(t_2)$ for all t_1 and t_2, with $a < t_1 < t_2 < b$; that is, C never intersects itself between its endpoints. The curve C is **closed** if $\mathbf{r}(a) = \mathbf{r}(b)$; that is, the initial and terminal points of C are the same.

Connected and Simply Connected Regions

An open region R in $\mathbb{R}^2$ (or D in $\mathbb{R}^3$) is **connected** if it possible to connect any two points of R by a continuous curve lying in R. An open region R is **simply connected** if every closed simple curve in R can be deformed and contracted to a point in R.

Conservative Vector Field

A vector field F is said to be **conservative** on a region (in $\mathbb{R}^2$ or $\mathbb{R}^3$) if there exists a scalar function φ such that $\mathbf{F} = \nabla\varphi$ on that region.

Test for Conservative Vector Fields

Let $\mathbf{F} = \langle f, g, h\rangle$ be a vector field defined on a connected and simply connected region D of $\mathbb{R}^3$, where f, g, and h have continuous first partial derivatives on D. Then $\mathbf{F}$ is a conservative vector field on D (there is a potential function φ such that $\mathbf{F} = \nabla\varphi$) if and only if

$$\frac{\partial f}{\partial y} = \frac{\partial g}{\partial x}, \quad \frac{\partial f}{\partial z} = \frac{\partial h}{\partial x}, \quad \text{and} \quad \frac{\partial g}{\partial z} = \frac{\partial h}{\partial y} \tag{18}$$

For vector fields in $\mathbb{R}^2$, we have the single condition $\frac{\partial f}{\partial y} = \frac{\partial g}{\partial x}$.

Finding Potential Functions in $\mathbb{R}^3$

Suppose $\mathbf{F} = \langle f, g, h\rangle$ is a conservative vector field. To find φ such that $\mathbf{F} = \nabla\varphi$, take the following steps:

1. Integral $\varphi_x = f$ with respect to x to obtain φ, which includes an arbitrary function $c(y, z)$.

2. Compute φ_y and equate it to g to obtain an expression for $c_y(y, z)$.

3. Integrate $c_y(y, z)$ with respect to y to obtain $c(y, z)$, including an arbitrary function $d(z)$.

4. Compute φ_z and equate it to h to get $d(z)$.

Beginning the procedure with $\varphi_y = g$ or $\varphi_z = h$ maybe be easier in some cases.

Fundamental Theorem for Line Integrals

Let $\mathbf{F}$ be a continuous vector field on an open connected region R in $\mathbb{R}^2$ (or D in $\mathbb{R}^3$). There exists a potential function φ with $\mathbf{F} = \nabla\varphi$ (which means that $\mathbf{F}$ is conservative) if and only if

$$\int\limits_C \mathbf{F} \cdot \mathbf{T}\, ds = \int\limits_C \mathbf{F} \cdot d\mathbf{r} = \varphi(B) - \varphi(A) \tag{19}$$

for all points A and B in R and all smooth oriented curves C from A to B.

Line Integrals on Closed Curves

Let R in $\mathbb{R}^2$ (or D in $\mathbb{R}^3$) be an open region. Then $\mathbf{F}$ is a conservative vector field on R if and only if $\oint_C \mathbf{F} \cdot d\mathbf{r} = 0$ on all simple closed smooth oriented curves C in R.

15.4 Green's Theorem

Let C be a simple closed smooth curve, oriented counterclockwise, that encloses a connected and simply connected region R in the plane. Assume $\mathbf{F} = \langle f, g \rangle$, where f and g have continuous first partial derivatives in R. Then

$$\oint_C \mathbf{F} \cdot d\mathbf{r} = \oint f\,dx + g\,dy = \iint_R \left(\frac{\partial g}{\partial x} - \frac{\partial f}{\partial y} \right) dA. \tag{20}$$

Two-Dimensional Curl

The **two-dimensional curl** of the vector field $\mathbf{F} = \langle f, g \rangle$ is $\frac{\partial g}{\partial x} - \frac{\partial f}{\partial y}$. If the curl is zero throughout a region, the vector field is said to be **irrotational** on that region.

Area of a Plane Region by Line Integrals

Under the conditions of Green's Theorem, the area of a region R enclosed by a curve C is

$$\oint_C x\,dy = -\oint_C y\,dx = \frac{1}{2} \oint_C (x\,dy - y\,dx) \tag{21}$$

Green's Thoerem, Flux Form

Let C be a simple closed smooth curve, oriented counterclockwise, that encloses a connected and simply connected region R in the plane. Assume $\mathbf{F} = \langle f, g \rangle$, where f and g have continuous first partial derivatives in R. Then

$$\oint_C \mathbf{F} \cdot \mathbf{n}\,ds = \oint_C f\,dy - g\,dx = \iint_R \left(\frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} \right) \tag{22}$$

where $\mathbf{n}$ is the outward unit normal vector on the curve.

Two-Dimensional Divergence

The **two-dimensional divergence** of the vector field $\mathbf{F} = \langle f,\, g\rangle$ is $\frac{\partial f}{\partial x} + \frac{\partial g}{\partial y}$. If the divergence is zero thoughout a region, the vector field is said to be **source free** on that region.

15.5 Divergence and Curl

Divergence of a Vector Field

The **divergence** of a vector field $\mathbf{F} = \langle f,\, g,\, h\rangle$ that is differentiable on a region of $\mathbb{R}^3$ is

$$\operatorname{div}\mathbf{F} = \nabla \cdot \mathbf{F} = \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} + \frac{\partial h}{\partial z} \tag{23}$$

If $\nabla \cdot \mathbf{F} = 0$, the vector field is **source free.**

Divergence of Radial Vector Fields

For a real number p, the divergence of the radial vector field

$$\mathbf{F} = \frac{\mathbf{r}}{|r|^p} = \frac{\langle x,\, y,\, z\rangle}{(x^2+y^2+z^2)^{\frac{p}{2}}} \text{ is } \nabla \cdot \mathbf{F} = \frac{3-p}{|\mathbf{r}|^p} \tag{24}$$

Curl of a Vector Field

The **curl** of a vector field $\mathbf{F} = \langle f,\, g,\, h\rangle$ that is differentiable on a region of $\mathbb{R}^3$ is

$$\nabla \times \mathbf{F} = \operatorname{curl}\mathbf{F} = \left(\frac{\partial h}{\partial y} - \frac{\partial g}{\partial z}\right)\hat{\mathbf{i}} + \left(\frac{\partial f}{\partial z} - \frac{\partial h}{\partial x}\right)\hat{\mathbf{j}} + \left(\frac{\partial g}{\partial x} - \frac{\partial f}{\partial y}\right)\hat{\mathbf{k}} \tag{25}$$

If $\nabla \times \mathbf{F} = \mathbf{0}$, the vector field is **irrotational**.

Curl of a Conservative Vector Field

The **general rotation vector field** is $\mathbf{F} = \mathbf{a} \times \mathbf{r}$, where the nonzero constant vector $\mathbf{a} = \langle a_1,\, a_2,\, a_3\rangle$ is the axis of ration and $\mathbf{r} = \langle x,\, y,\, z\rangle$. For all nonzero choices of $\mathbf{a}$, $|\nabla \times \mathbf{F}| = 2|\mathbf{a}|$ and $\nabla \cdot \mathbf{F} = 0$. The constant angular speed of the vector field is

$$\omega = |\mathbf{a}| = \frac{1}{2}|\nabla \times \mathbf{F}| \tag{26}$$

Curl of a Conservative Vector Field

Suppose that $\mathbf{F}$ is a conservative vector field on an open region D of $\mathbb{R}^3$. Let $\mathbf{F} = \nabla\varphi$, where φ is a potential function with continuous second partial derivatives on D. Then $\nabla \times \mathbf{F} = \nabla \times \nabla\varphi = \mathbf{0}$; that is, the curl of the gradient is the zero vector and $\mathbf{F}$ is irrotational.

Divergence of the Curl

Suppose that $\mathbf{F} = \langle f, g, h\rangle$, where f, g, and h have continuous second partial derivatives. Then $\nabla \cdot (\nabla \times \mathbf{F}) = 0$: The divergence of the curl is zero.

Product Rule for the Divergence

Let u be a scalar-valued functino that is differentiable on a region D and let $\mathbf{F}$ be a vector field that is differentiable on D. Then

$$\nabla \cdot (u\mathbf{F}) = \nabla u \cdot \mathbf{F} + u(\nabla \cdot \mathbf{F}) \tag{27}$$

Properties of a Conservative Vector Field

Let $\mathbf{F}$ be a conservative vector field whose components have continuous second partial derivatives on an open connected region D in $\mathbb{R}^3$. Then $\mathbf{F}$ has the following equivalent properties.

1. There exists a potential function φ such that $\mathbf{F} = \nabla\varphi$
2. $\int_C \mathbf{F} \cdot d\mathbf{r} = \varphi(B) - \varphi(A)$ for all points A and B in D and all smooth oriented curves C from A and B.
3. $\oint_C \mathbf{F} \cdot d\mathbf{r} = 0$ on all simple smooth closed oriented curves C in D.
4. $\nabla \times \mathbf{F} = \mathbf{0}$ at all points of D.

15.6 Surface Integrals

Surface Integrals of Scalar-Valued Functions on Parameterized Surface

Let f be a continuous function on a smooth surface S given parametrically by $\mathbf{r}(u,\, v) = \langle x(u,\, v),\, y(u,\, v),\, z(u,\, v)\rangle$, where $R = \{(u,\, v) : a \le u \le b, c \le v \le d\}$. Assume also that the tangent vectors $\mathbf{t}_u = \frac{\partial \mathbf{r}}{\partial u} = \langle \frac{\partial x}{\partial u},\, \frac{\partial y}{\partial u},\, \frac{\partial z}{\partial u}\rangle$, and $\mathbf{t}_v = \frac{\partial \mathbf{r}}{\partial v} = \langle \frac{\partial x}{\partial v},\, \frac{\partial y}{\partial v},\, \frac{\partial z}{\partial v}\rangle$ are continuous on R and the normal vectors $\mathbf{n} = \mathbf{t}_u \times \mathbf{t}_v$ is nonzero on R. Then the **surface integral** of the scalar-valued function f over S is

$$\iint_S f(x,\, y,\, z)\, dS = \iint_R f(x(u,\, v),\, y(u,\, v),\, z(u,\, v))|\mathbf{t}_u \times \mathbf{t}_v|\, dA \tag{28}$$

If $f(x,\, y,\, z) = 1$, the integral equals the surface area of S.

Evaluation of Surface Integrals of Scalar-Valued Functions on Explicitly Defined Surfaces

Let f be a continuous function on a smooth surfaces S given by $z = g(x,\, y)$, for $(x,\, y)$ in a region R. The surface integral of f over S is

$$\iint_S f(x,\, y,\, z)\, dS = \iint_R f(x,\, y,\, g(x,\, y))\sqrt{z_x^2 + z_y^2 + 1}\, dA \tag{29}$$

If $f(x,\, y,\, z) = 1$, the surface integral equals the area of the surface.

Surface Integral of a Vector Field

Suppose $\mathbf{F} = \langle f,\, g,\, h\rangle$ is a continuous vector field on a region of $\mathbb{R}^3$ containing a smooth oriented surface S. If S is defined parametrically as $\mathbf{r}(u,\, v) = \langle x(u,\, v),\, y(u,\, v),\, z(u,\, v)\rangle$, for $(u,\, v)$ is a region R, then

$$\iint_S \mathbf{F} \cdot \mathbf{n}\, dS = \iint_R \mathbf{F} \cdot (\mathbf{t}_u \times \mathbf{t}_v)\, dA \tag{30}$$

where $\mathbf{t}_u = \frac{\partial \mathbf{r}}{\partial u} = \langle \frac{\partial x}{\partial u},\, \frac{\partial y}{\partial u},\, \frac{\partial z}{\partial u}\rangle$ and $\mathbf{t}_v = \frac{\partial \mathbf{r}}{\partial v} = \langle \frac{\partial x}{\partial v},\, \frac{\partial y}{\partial v},\, \frac{\partial z}{\partial v}\rangle$ are continuous on R, the normal vector $\mathbf{n} = \mathbf{t}_u \times \mathbf{t}_v$ is nonzero on R, and the direction of $\mathbf{n}$ is consistent with the orientation of S. If S is defined in the form $z = g(z,\, y)$, for $(x,\, y)$ in a region R, then

$$\iint\limits_S \mathbf{F} \cdot \mathbf{n}\, dS = \iint\limits_R (-f z_x - g z_y + h)\, dA \tag{31}$$

15.7 Stokes' Theorem

Let S be a smooth oriented surface in $\mathbb{R}^2$ with a smooth closed boundary C whose orientation is consistent with that of S. Assume that $\mathbf{F} = \langle f,\, g,\, h\rangle$ is a vector field whose components have continuous first partial derivatives on S. Then

$$\oint_C \mathbf{F} \cdot d\mathbf{r} = \iint_S (\nabla \times \mathbf{F}) \cdot \mathbf{n}\, dS \tag{32}$$

where $\mathbf{n}$ is the unit vector normal to S determined by the orientation of S.

Curl $\mathbf{F} = \mathbf{0}$ Implies $\mathbf{F}$ is Conservative

Suppose that $\nabla \times \mathbf{F} = \mathbf{0}$ throughout an open simply connected region D of $\mathbb{R}^3$. Then $\oint_C \mathbf{F} \cdot d\mathbf{r} = 0$ on all closed simple smooth curves C in D and $\mathbf{F}$ is a conservative vector field on D.

15.8 Divergence Theorem

Let $\mathbf{F}$ be a vector field whose components have continuous first partial derivatives in a connected and simply connected region D in $\mathbb{R}^3$ enclosed by a smooth oriented surface S. Then

$$\iint\limits_S \mathbf{F} \cdot \mathbf{n}\, dS = \iiint\limits_D \nabla \cdot \mathbf{F}\, dV \tag{33}$$

where $\mathbf{n}$ is the unit outward normal vector on S.

Divergence Theorem for Hollow Regions

Suppose the vector field $\mathbf{F}$ satisfies the conditions of the Divergence Theorem on a region D bounded by two smooth oriented surfaces S_1 and S_2, where S_1 lies within S_2. Let S be the entire boundary of $D(S = S_1 \cup S_2)$ and let $\mathbf{n}_1$ and $\mathbf{n}_2$ be the outward unit normal vectors for S_1 and S_2, respectively. Then

$$\iiint\limits_D \nabla \cdot \mathbf{F}\, dV = \iint \mathbf{F} \cdot \mathbf{n} dS = \iint\limits_{S_2} \mathbf{F} \cdot \mathbf{n_2}\, dS = \iint\limits_{S_1} \mathbf{F} \cdot \mathbf{n_1}\, dS \tag{34}$$

Appendix A

Build System Documentation

This portfolio was compiled using a custom build system that:

- Uses XeLaTeX for modern font support
- Automatically discovers and builds LaTeX documents
- Excludes incomplete templates and problematic files
- Preserves original document formatting and styles
- Supports both PDF inclusion and LaTeX compilation

The build system makefile identifies 91 buildable LaTeX documents across the academic portfolio while maintaining the integrity of each individual document's styling and content.

Appendix B

Course Categories

Foundational Courses (CS 1000-2000)

Basic programming, discrete mathematics, and computer science foundations.

Systems & Algorithms (CS 2000-4000)

Data structures, algorithms, operating systems, and software engineering.

Advanced & Graduate (CS 5000+)

Advanced topics in artificial intelligence, numerical modeling, and algorithm analysis.

Mathematics

Calculus sequence, foundations of mathematics, and applied mathematics.

Engineering & Physics

Computer engineering, embedded systems, and physics coursework.