DegreeLive

BBA	SEMESTER V	UNIVERSITY OF KERALA	COURSE CODE: UK5DSCMGT303

MANAGEMENT SCIENCE

EXAM REVISION NOTES

JOIN WHATSAPP CHANNEL →

MODULE I

INTRODUCTION TO MANAGEMENT SCIENCE, MODELLING, AND DECISION ENVIRONMENTS

Foundations of Management Science and Operations Research: Management Science, historically designated as Operations Research (OR), represents the systematic application of advanced scientific methodology, mathematical modeling, statistical analysis, and algorithmic optimization to resolve complex decision problems arising in the direction and management of large systems of men, machines, materials, money, and information. The discipline originated during the Second World War in 1937 at the Bawdsey Research Station in Britain, where Nobel physicist P. M. S. Blackett assembled a multidisciplinary scientific team known as "Blackett's Circus" to optimize radar deployment, anti-aircraft coordination, and convoy logistics. Following 1945, these quantitative techniques diffused rapidly into commercial industrial administration, catalyzed by George Dantzig's invention of the Simplex Algorithm in 1947.

Methodological Problem-Solving Phases: The application of Management Science follows a rigorous six-phase cycle: **Phase 1: Problem Definition:** Observing operational systems, engaging key stakeholders, establishing system boundaries, and isolating root causes from superficial symptoms. **Phase 2: Mathematical Model Formulation:** Translating verbal narratives into symbolic equations, identifying controllable decision variables, formalizing the objective function, and establishing structural constraints. **Phase 3: Model Solution & Optimization:** Executing mathematical algorithms or heuristic solvers to determine optimal resource allocations. **Phase 4: Model Validation:** Back-testing model predictions against historical performance records to confirm fidelity to operational reality. **Phase 5: Sensitivity Analysis:** Evaluating how changes in input parameters, cost coefficients, and constraint capacities affect the optimal solution. **Phase 6: Implementation & Control:** Translating mathematical results into standard managerial operating procedures and monitoring long-term performance.

Taxonomy of Management Models: A model is an abstraction of reality constructed to analyze system behavior without physical disruption. Models are classified across four dimensions: **Physical Structure: Iconic Models:** Scaled visual physical replicas (e.g., architectural blueprints, 3D factory layout prototypes). **Analogue Models:** Physical representations behaving like the system without visual resemblance (e.g., electrical circuit graphs, mercury thermometers). **Symbolic Models:** Alphanumeric mathematical equations and inequalities representing system logic. **Certainty of Parameters: Deterministic Models:** All coefficients, demands, and constraints are known with absolute certainty. **Stochastic Models:** Incorporate random probability distributions to account for real-world variance. **Time Dimension: Static Models:** Analyze a single fixed point in time. **Dynamic Models:** Incorporate time as an active variable where current decisions alter future states. **Normative Scope: Descriptive Models:** Describe how a system functions without claiming optimality. **Prescriptive Models:** Prescribe the unique mathematically optimal course of action.

Decision Making Under Uncertainty — Five Classical Criteria: In environments where future states of nature cannot be assigned probabilities, decision-makers deploy non-probabilistic criteria: **Maximin Criterion (Wald's Rule of Pessimism):** Assumes nature is actively hostile; identifies the minimum payoff for each alternative and selects the alternative that maximizes these minimums. **Maximax Criterion (Rule of Extreme Optimism):** Seeks the maximum potential payoff regardless of downside risks; selects the alternative with the highest individual payoff. **Minimax Regret (Savage Opportunity Loss Criterion):** Formulates a Regret Matrix by subtracting each cell from its column maximum payoff; selects the alternative that minimizes maximum regret. **Hurwicz Criterion (Alpha Index):** Balances optimism and pessimism using a coefficient of realism alpha between 0 and 1: Hurwicz Index = alpha multiplied by maximum payoff plus (1 - alpha) multiplied by minimum payoff. **Laplace Criterion (Equal Likelihood):** Applies the Principle of Insufficient Reason, assigning equal probability (1/n) to all states of nature and selecting the alternative with the highest arithmetic mean payoff.

Decision Making Under Risk — EMV and EVPI: When valid probabilities exist for each state of nature: **Expected Monetary Value (EMV):** The weighted average payoff across all states: EMV equals the sum of (probability multiplied by payoff). **Expected Value of Perfect Information (EVPI):** The maximum theoretical price an enterprise should pay to eliminate uncertainty: EVPI = EPPI - EMV*, where EPPI is Expected Payoff with Perfect Information. **Decision Tree Analysis:** Graphical representation of sequential, multi-stage decisions using Decision Nodes (squares) where managers choose alternatives, and Chance Nodes (circles) where probabilistic states of nature occur; solved via **Backward Induction (Rollback Method)** by computing EMVs at chance nodes and selecting optimal branches at decision nodes from right to left.

Decision Support Systems (DSS) and Quantitative Ethics: Modern management science is operationalized via enterprise DSS consisting of: **Database Management Subsystem (DBMS):** Extracts real-time operational data from ERP systems. **Model Base Management Subsystem (MBMS):** Stores optimization algorithms, LP solvers, and forecasting models. **Dialogue Subsystem:** Visual interactive interface for executive "What-If" scenario modeling. Quantitative modelers must guard against cognitive biases—including anchoring bias, overconfidence, and confirmation bias—and incorporate environmental and human welfare constraints into objective functions.

Utility Theory and Managerial Risk Attitudes: When financial stakes are exceptionally large relative to enterprise capitalization, expected monetary value fails because decision-makers exhibit non-linear utility curves: **Risk Averse Decision Makers:** Possess concave utility functions displaying diminishing marginal utility of wealth; they accept guaranteed lower payoffs (risk premiums) to avoid high-consequence downside losses. **Risk Neutral Decision Makers:** Possess linear utility functions where utility is strictly proportional to money; their decisions align identically with EMV criteria. **Risk Seeking Decision Makers:** Possess convex utility functions; they are willing to accept actuarially unfavorable expected values in pursuit of extraordinary jackpot returns.

DegreeLive

MODULE II

LINEAR PROGRAMMING PROBLEMS, GRAPHICAL OPTIMIZATION, AND SPECIAL CASES

Linear Programming (LP) Core Architecture: Linear programming is a mathematical optimization technique designed to allocate scarce organizational resources among competing productive activities. A standard LP model comprises: **Decision Variables:** Controllable quantities to be determined (x1, x2). **Objective Function:** Linear mathematical equation to be maximized (profit/revenue) or minimized (cost/time). **Structural Constraints:** System of linear inequalities reflecting physical, labor, or capacity limits. **Non-Negativity Restrictions:** Explicit requirement that decision variables must be non-negative (x1, x2 >= 0).

Foundational LP Assumptions: Linearity (Proportionality): Resource consumption and objective contribution are strictly proportional to activity volume, ruling out economies of scale. **Additivity:** Total payoff and total resource usage equal the algebraic sum of individual contributions, ruling out cross-product cannibalization. **Divisibility:** Decision variables are continuous real numbers admitting fractional values. **Determinism:** All coefficients, capacities, and costs are known with certainty. **Finiteness:** Number of variables and constraints is finite.

Graphical Solution Algorithm for Two-Variable LP: The graphical method applies to two-variable models based on the **Fundamental Theorem of LP**, which proves that the optimal solution will always lie at an extreme corner vertex of the convex feasible polygon. **Step 1:** Plot non-negativity axes in the positive first quadrant. **Step 2:** Convert structural inequalities into equality boundary lines by calculating axis intercepts. **Step 3:** Identify the common feasible region satisfying all inequalities simultaneously. **Step 4:** Determine coordinates of all corner vertices via simultaneous linear equations. **Step 5:** Evaluate the objective function at each corner vertex; select the vertex yielding maximum profit or minimum cost.

Iso-Profit and Iso-Cost Contour Optimization: In addition to evaluating discrete corner points, graphical optimization utilizes the **Iso-Profit Line Method**: an arbitrary profit line (c1 x1 + c2 x2 = Z0) is established and shifted parallelly away from the coordinate origin. The slope of the iso-profit line equals -c1 / c2. The optimal solution is the absolute last extreme corner vertex of the feasible polygon that the expanding iso-profit line touches before completely leaving the feasible space. In cost minimization problems, the iso-cost line is shifted inwards toward the origin until it touches the closest feasible boundary vertex.

Special Computational Cases in Linear Programming: Multiple (Alternative) Optimal Solutions: Occurs when the slope of the objective function is identical to the slope of a binding constraint boundary, causing the iso-profit line to coincide along an entire boundary segment; any point on the segment achieves optimal profit. **Infeasible Solution:** Arises when constraints are mutually contradictory with no overlapping area; intersection of half-planes is empty. **Unbounded Solution:** Occurs when the feasible region is open-ended in the direction of optimization, allowing objective value to grow infinitely without violating constraints; indicates missing limiting constraints. **Redundant Constraint:** A constraint that does not restrict the feasible region and can be eliminated without altering the optimal solution.

Simplex Method Foundations and Standard Form: When linear programming problems involve three or more decision variables, graphical geometry fails, necessitating the algebraic **Simplex Algorithm**. Problems are converted into **Standard Form** by converting inequalities into strict equalities: **Slack Variables (s_i >= 0):** Added to less-than-or-equal-to constraints to represent unutilized, idle resource capacity. **Surplus Variables (e_i >= 0):** Subtracted from greater-than-or-equal-to constraints to represent activity exceeding minimum requirements. **Artificial Variables (A_i >= 0):** Added to equality and surplus constraints to form an initial identity basis; penalized heavily in the objective function (via Big-M or Two-Phase method) to force them to zero in the final optimal solution. **Basic Feasible Solution (BFS):** In a system of m equations and n variables, set n - m non-basic variables to zero and solve for the remaining m basic variables; every BFS corresponds algebraically to an extreme corner point of the feasible polygon.

Duality and Economic Shadow Prices: Every primal linear program has an associated dual problem. The decision variables of the dual problem are **Shadow Prices**, representing the marginal increase in the objective function value achievable by relaxing a specific resource constraint by one unit. Non-binding constraints with unused slack possess a shadow price of identically zero; binding bottleneck constraints possess positive shadow prices reflecting the maximum economic premium management should pay for additional capacity.

MODULE III

ASSIGNMENT PROBLEMS, THE HUNGARIAN METHOD, AND SPECIAL VARIATIONS

Nature of the Assignment Problem: A specialized sub-class of linear transportation problems where n facilities (workers, machines) are allocated to n destinations (jobs, tasks) on a **strict one-to-one basis** at minimum aggregate cost or maximum revenue. Characterized by binary decision variables: x_ij equals 1 if facility i is assigned to job j, and 0 otherwise. Standard LP and transportation algorithms fail due to extreme degeneracy, as an assignment requires exactly n allocations while transportation BFS requires 2n - 1 basic cells.

The Hungarian Method (Kuhn-Munkres Algorithm): Based on **König's Theorem**, which establishes that adding or subtracting a constant to any row or column of a cost matrix leaves the optimal assignment schedule unchanged. **Step 1: Row Reduction:** Subtract the minimum element of each row from all elements in that row. **Step 2: Column Reduction:** Subtract the minimum element of each column from all elements in that column. **Step 3: Line Covering Test:** Draw the minimum number of horizontal and vertical straight lines (k) necessary to cover all zeros in the reduced matrix. If k = n, an optimal zero assignment exists; proceed to Step 5. **Step 4: Matrix Augmentation:** If k < n, find the smallest uncovered element (u); subtract u from all uncovered elements, add u to all intersection elements, and leave single-covered elements unchanged; return to Step 3. **Step 5: Zero Assignment Protocol:** Identify rows and columns with single zeros, box the assigned zero, and cross out competing zeros in that row/column. **Step 6: Optimal Cost Calculation:** Sum the original unreduced costs corresponding to boxed assignments.

DegreeLive

Special Variations in Assignment Problems: Unbalanced Assignment: When workers do not equal jobs (m != n); balanced by introducing Dummy Rows or Dummy Columns with zero costs. Allocating to a dummy column represents an idle worker; allocating to a dummy row represents an unfulfilled job that must be outsourced. **Maximization Assignment:** When matrix entries represent revenues or efficiencies; converted to minimization by finding the global maximum element and subtracting every cell from it to create a Relative Opportunity Loss (Regret) matrix. **Restricted Assignments:** Prohibited allocations due to safety, licensing, or union rules are assigned a prohibitive penalty cost M approaching infinity. **Traveling Salesman Problem (TSP):** A routing assignment problem where a salesperson visits n cities exactly once and returns home; requires Sub-tour Elimination Constraints to prevent isolated cyclic loops.

Operational Comparison: Assignment versus Transportation Problems: Assignment: Bounded by strict 1-to-1 matching; square matrix (n x n); exact supply = 1, exact demand = 1; exactly n positive basic allocations in final schedule; solved via Hungarian method. **Transportation:** Deals with aggregate commodity volumes; rectangular matrix (m x n); variable origin supply (a_i) and destination demand (b_j); requires exactly m + n - 1 positive allocations; solved via VAM and MODI.

MODULE IV

TRANSPORTATION PROBLEMS, INITIAL FEASIBLE SOLUTIONS, AND MODI OPTIMALITY

Transportation Model Architecture: A linear distribution model designed to allocate homogeneous goods from m supply origins (capacities a_i) to n demand destinations (requirements b_j) at minimum total freight cost. **Balanced Transportation:** Total supply equals total demand. **Unbalanced Transportation:** Total supply does not equal total demand; balanced by inserting a Dummy Destination (with supply deficit) or Dummy Origin (with demand deficit) with zero freight rates. **Basic Feasible Solution Property:** Requires exactly **m + n - 1** non-negative allocations occupying independent, loopless cells.

Three Classical Methods for Initial Basic Feasible Solution (IBFS): North West Corner Rule (NWCR): Purely positional heuristic allocating maximum feasible quantity to top-left cell (1, 1), exhausting supply or demand, and stepping right or down; completely ignores freight costs, yielding poor starting solutions. **Lowest Cost Method (LCM / Matrix Minimum):** Greedy economic heuristic allocating maximum feasible volume to the absolute lowest cost cell in the entire matrix; yields lower costs than NWCR but can force expensive late allocations. **Vogel's Approximation Method (VAM):** Best-in-class penalty regret heuristic; calculates the difference between the two lowest costs in each row and column, identifies the vector with the highest penalty, and allocates to its minimum cost cell; frequently yields solutions within 2% of global optimality.

Optimality Testing via Modified Distribution (MODI / u-v Method): Evaluates whether an IBFS can be improved. **Step 1: Dual Variables:** Set u1 = 0 and solve u_i + v_j = c_ij for all allocated basic cells. **Step 2: Opportunity Cost Evaluation:** For all empty unallocated cells, calculate Net Evaluation Index: Delta_ij = c_ij - (u_i + v_j). **Optimality Criterion:** If all Delta_ij >= 0, the solution is GLOBALLY OPTIMAL! If any Delta_ij < 0, total freight cost can be reduced.

Stepping Stone Reallocation Loop: If Delta_ij < 0, select the most negative cell as entering cell. Form a closed loop of alternating horizontal and vertical line segments turning exclusively at basic cells. Assign alternating (+, -) signs starting with (+) at entering cell. Identify the minimum allocation among negative corners (theta); add theta to positive corners and subtract theta from negative corners to establish an improved distribution schedule.

Degeneracy Resolution in Transportation: Occurs when basic allocations are strictly less than m + n - 1, making dual variable calculations impossible. Resolved via the **Epsilon Protocol**: allocate an infinitesimal quantity (epsilon > 0) to an unallocated cell with low cost that does not form a closed loop; epsilon is treated as a basic allocation during dual calculations, but ignored in physical freight sums.

Transshipment Problems and Intermediate Hub Logistics: Classical transportation models assume goods move directly from plants to retail depots. In contemporary supply networks, physical freight frequently transits through intermediate transshipment hubs (cross-docking terminals, regional consolidation centers). Transshipment models convert multi-tiered networks into equivalent single-stage transportation tableaus by assigning a large buffer capacity (equal to total system volume) to all intermediate transshipment nodes.

MODULE V

PROJECT MANAGEMENT, NETWORK ANALYSIS (CPM & PERT), AND PROJECT CRASHING

Project Fundamentals and Network Topologies: A project is a non-repetitive, temporary endeavor with a defined lifecycle bound by the Triple Constraint of Scope, Time, and Cost. Workflows are modeled using directed graphs: **Activity-on-Arrow (AOA):** Arrows represent activities; events are circular nodes; requires dotted dummy activities with zero duration to preserve logic. **Activity-on-Node (AON):** Nodes represent activities; arrows indicate dependencies; eliminates dummy activities. **Fulkerson's Rule:** Systematic numbering of events ensuring tail number is strictly less than head number (i < j). **Topologic Errors:** Looping (cycles), Dangling (disconnected ends), and redundant dummies must be strictly eliminated.

Critical Path Method (CPM) for Deterministic Networks: Assumes activity durations are known with certainty. **Forward Pass:** Computes Earliest Start (ES) and Earliest Finish (EF = ES + t); at merge events, E_j = max(EF of incoming activities). **Backward Pass:** Computes Latest Finish (LF) and Latest Start (LS = LF - t); at burst events, L_i = min(LS of outgoing activities). **Critical Path:** The longest continuous sequence of activities from start to finish with zero total float; determines minimum project completion duration.

Float (Slack) Metrics: Total Float (TF): Maximum delay permitted without extending overall project completion: TF = LS - ES = LF - EF. **Free Float (FF):** Delay permitted without delaying earliest start of succeeding activities: FF = E_j - EF. **Independent Float (IF):** Delay permitted under worst-case boundary conditions: IF = max[0, E_j - L_i - t]. **Interfering Float (ITF):** Portions of total float that delay successor starts: ITF = TF - FF.

Programme Evaluation and Review Technique (PERT): Designed for non-deterministic R&D projects with uncertain durations. Assumes a continuous Beta distribution parameterized by three time estimates: **Optimistic Time (a):** Minimum duration under ideal conditions. **Most Likely Time (m):** Modal duration under typical conditions. **Pessimistic Time (b):** Maximum duration under adverse conditions. **Expected Duration:** t_e = (a + 4m + b) / 6. **Activity Variance:** sigma^2 = ((b - a) / 6)^2. **Project Variance:** Sum of variances of critical path activities only. **Normal Deviate:** Z = (T_s - T_e) / sigma_P, used to determine probability of meeting target deadline T_s via cumulative normal distribution tables.

Project Crashing and Time-Cost Trade-Offs: Compressing project duration involves balancing increasing direct costs (overtime, extra shifts) against decreasing indirect overhead costs. **Cost Slope Formula:** S = (Crash Cost - Normal Cost) / (Normal Time - Crash Time). **Crashing Protocol:** Crash the critical activity possessing the lowest cost slope up to its crash limit or until a parallel path becomes critical; crash parallel critical paths simultaneously until total project costs reach the global minimum (Optimum Project Duration).

Resource Allocation, Leveling, and Smoothing: Resource Histogram: Graphical profile of daily labor or machine consumption across project calendar time. **Resource Smoothing:** Shifts non-critical activities within their available total float to eliminate sharp demand peaks without altering project completion date. **Resource Leveling:** Delays activities beyond available float when daily resource availability is strictly capped, extending overall project completion duration.

Earned Value Management (EVM) and S-Curves: Project tracking via three core metrics: **Planned Value (PV):** Budgeted cost of scheduled work. **Earned Value (EV):** Budgeted cost of work performed. **Actual Cost (AC):** Realized cost incurred. Schedule Variance SV = EV - PV; Cost Variance CV = EV - AC. Schedule Performance Index SPI = EV / PV; Cost Performance Index CPI = EV / AC. Estimate at Completion EAC = BAC / CPI. To-Complete Performance Index TCPI = (BAC - EV) / (BAC - AC). **Cumulative S-Curves:** Graphical trajectory of cumulative project expenditures displaying slow startup, steep execution acceleration, and tapering closeout.

Comparative Architecture: CPM versus PERT: CPM: Deterministic single time estimate, activity-oriented, directly integrates time-cost crashing, best suited for recurring civil construction and plant overhauls. **PERT:** Probabilistic three time estimates modeled via Beta distribution, event-oriented, focuses on schedule variance and completion probability, best suited for non-repetitive R&D, defense, and pharmaceutical clinical trials.

Work Breakdown Structure (WBS) Five-Tier Hierarchy: Level 1 (Project): Total project scope deliverable. **Level 2 (Sub-projects):** Major geographic or engineering phases. **Level 3 (Systems):** Functional engineering subsystems. **Level 4 (Work Packages):** Discrete cost accounts with assigned control account managers. **Level 5 (Activities):** Granular work tasks mapped directly to CPM/PERT network arrows.

DegreeLive