```
%==========================================================================
% MATLAB Simulation and Analysis for Multistage Rockets
%
% This script performs two main analyses:
% 1. Practical: Compares the theoretical optimum with real-world data from
%    the SpaceX Falcon 9 rocket to illustrate engineering trade-offs.
%
%==========================================================================

clear;
clc;
close all;

% Define universal constants
g0 = 9.80665; % Standard gravity in m/s^2
```

Theory vs. Reality - Falcon 9 Case Study

This section compares the theoretical model against real-world data for the SpaceX Falcon 9 rocket.

```
%--------------------------------------------------------------------------

fprintf('====================================================\n');
fprintf('Part 2: Theory vs. Reality - Falcon 9 Case Study\n');
fprintf('====================================================\n');

% --- Step 1: Define the REAL Falcon 9 Parameters (from public data) ---
F9.payload_kg = 16800;
F9.s1_dry_mass = 25600;
F9.s1_prop_mass = 395700;
F9.s2_dry_mass = 3900;
F9.s2_prop_mass = 92670;
F9.Isp = [300, 348]; % [Stage 1 avg, Stage 2 vacuum]

% --- Step 2: Calculate the "Real World" Metrics from the data ---
F9.s1_total_mass = F9.s1_dry_mass + F9.s1_prop_mass;
F9.s2_total_mass = F9.s2_dry_mass + F9.s2_prop_mass;
F9.liftoff_mass = F9.s1_total_mass + F9.s2_total_mass + F9.payload_kg;
F9.epsilon = [F9.s1_dry_mass / F9.s1_total_mass, F9.s2_dry_mass /
F9.s2_total_mass];
F9.Gamma = F9.payload_kg / F9.liftoff_mass;

% Estimate the ideal Delta-V produced by the Falcon 9 using Tsiolkovsky's eq.
M01 = F9.liftoff_mass;
Mf1 = M01 - F9.s1_prop_mass;
DeltaV1 = g0 * F9.Isp(1) * log(M01 / Mf1);
M02 = Mf1 - F9.s1_dry_mass;
Mf2 = M02 - F9.s2_prop_mass;
DeltaV2 = g0 * F9.Isp(2) * log(M02 / Mf2);
F9.DeltaV_produced = DeltaV1 + DeltaV2;

fprintf('--- Real-World Falcon 9 Analysis ---\n');
fprintf('Estimated Total Delta-V Produced: %.f m/s\n', F9.DeltaV_produced);
fprintf('Actual Overall Payload Fraction: %.4f (%.2f%%)\n', F9.Gamma,
F9.Gamma*100);
fprintf('Stage 1 Epsilon: %.3f, Stage 2 Epsilon: %.3f\n\n', F9.epsilon(1),
F9.epsilon(2));

% --- Step 3: Use our code to find the THEORETICAL OPTIMUM for this rocket ---
n_stages = 2;
[~, ~, ~, Gamma_optimal] = optimizeRocket(F9.DeltaV_produced, n_stages,
F9.Isp, F9.epsilon, g0);
fprintf('--- Theoretical Optimum for Falcon 9 Technology ---\n');
fprintf('For the same Delta-V and technology, the optimal payload fraction
is: %.4f (%.2f%%)\n\n', ...
    Gamma_optimal, Gamma_optimal*100);

% --- Step 4: Visualize the Comparison ---

% Plot 1: Bar chart comparing mass breakdowns
figure('Name', 'Mass Breakdown: Real vs. Optimal', 'Position', [150, 150,
900, 600]);

% Data for the bar chart [rows = groups, columns = categories]
real_masses_row = [F9.payload_kg, F9.s1_dry_mass + F9.s2_dry_mass,
F9.s1_prop_mass + F9.s2_prop_mass] / 1000; % in tonnes

optimal_payload_mass = Gamma_optimal * F9.liftoff_mass;
non_payload_mass = F9.liftoff_mass - optimal_payload_mass;
real_total_stage_mass = F9.s1_total_mass + F9.s2_total_mass;
real_total_structure_mass = F9.s1_dry_mass + F9.s2_dry_mass;
structure_fraction_of_stages = real_total_structure_mass /
real_total_stage_mass;
optimal_structure_mass = non_payload_mass * structure_fraction_of_stages;
optimal_propellant_mass = non_payload_mass * (1 -
structure_fraction_of_stages);
optimal_masses_row = [optimal_payload_mass, optimal_structure_mass,
optimal_propellant_mass] / 1000; % in tonnes

bar_data = [real_masses_row; optimal_masses_row];

b = bar(bar_data, 'grouped');
b(1).FaceColor = '#0072BD'; % Blue for Payload
b(2).FaceColor = '#D95319'; % Orange for Structure
b(3).FaceColor = '#77AC30'; % Green for Propellant

grid on;
title('Mass Breakdown Comparison (Assuming 549 Tonne Liftoff Mass)');
ylabel('Mass (tonnes)');
xticks(1:2);
xticklabels({'Real Falcon 9', 'Theoretically Optimal Rocket'});
legend('Payload Mass', 'Structural Mass', 'Propellant Mass', 'Location',
'northeast');
set(gca, 'FontSize', 12);

% Plot 2: Performance curve showing where Falcon 9 sits
```

```
figure('Name', 'Performance Curve: Real vs. Optimal', 'Position', [200, 200,
900, 600]);
hold on;
Vn_range = linspace(6000, 10000, 100);
Gamma_curve = zeros(size(Vn_range));
for i = 1:length(Vn_range)
    try
        [~, ~, ~, Gamma_curve(i)] = optimizeRocket(Vn_range(i), n_stages,
F9.Isp, F9.epsilon, g0);
    catch
        Gamma_curve(i) = NaN; % Mark as not a number if impossible
    end
end
plot(Vn_range, Gamma_curve * 100, 'b-', 'LineWidth', 2);
plot(F9.DeltaV_produced, F9.Gamma * 100, 'rx', 'MarkerSize', 12, 'LineWidth',
2);
grid on;
title('Performance: Theoretical Optimum vs. Real Falcon 9');
xlabel('Total Delta-V (m/s)');
ylabel('Overall Payload Fraction (Γ) [%]');
legend('Optimal Performance Curve (Pareto Front)', 'Actual Falcon 9
Performance', 'Location', 'northwest');
set(gca, 'FontSize', 12);
hold off;

fprintf('Part 2 plots generated successfully.\n');
```

```
====================================================
Part 2: Theory vs. Reality - Falcon 9 Case Study
====================================================
--- Real-World Falcon 9 Analysis ---
Estimated Total Delta-V Produced: 9767 m/s
Actual Overall Payload Fraction: 0.0314 (3.14%)
Stage 1 Epsilon: 0.061, Stage 2 Epsilon: 0.040
```

==

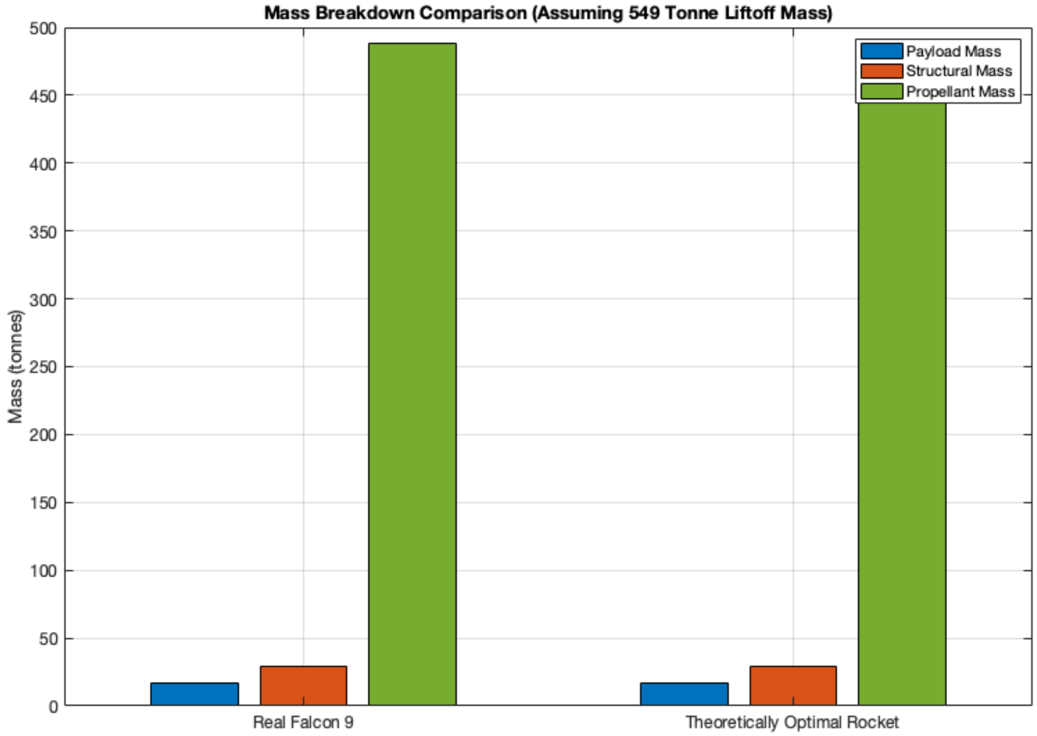
Function Definition: General Rocket Optimization
==

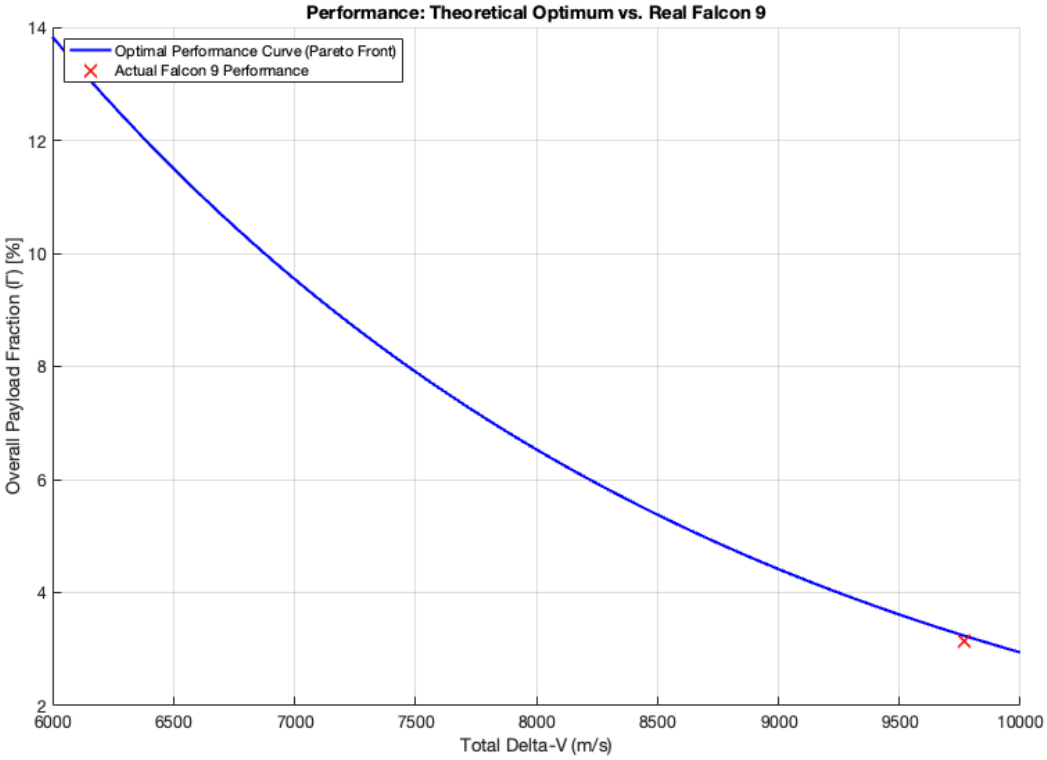
```
function [alpha, lambda, R, Gamma] = optimizeRocket(Vn_req, n, Isp, epsilon,
g0)
    % Implements the general optimization for a multistage rocket.
    if isscalar(Isp), Isp = ones(1, n) * Isp; elseif length(Isp) ~= n,
error('Isp vector must have length n or be a scalar.'); end
    if isscalar(epsilon), epsilon = ones(1, n) * epsilon; elseif
length(epsilon) ~= n, error('epsilon vector must have length n or be a
scalar.'); end
    C = Isp * g0;
    Vn_max_possible = sum(C .* log(1 ./ epsilon));
    if Vn_req >= Vn_max_possible
```

```
        error('Required Delta-V (%.f m/s) is >= max possible (%.f m/s)',
Vn_req, Vn_max_possible);
    end
    alpha_equation = @(a) sum(C .* log((C - a) ./ (epsilon .* C))) - Vn_req;
    upper_bound = min(C .* (1 - epsilon));
    search_interval = [0, upper_bound - 1e-6];
    options = optimset('Display','off');
    alpha = fzero(alpha_equation, search_interval, options);
    lambda = (alpha .* epsilon) ./ (C - C .* epsilon - alpha);
    R = (1 + lambda) ./ (epsilon + lambda);
    Gamma = prod(lambda ./ (1 + lambda));
end
```

```
--- Theoretical Optimum for Falcon 9 Technology ---
For the same Delta-V and technology, the optimal payload fraction is: 0.0324
(3.24%)

Part 2 plots generated successfully.
```





Published with MATLAB® R2024b