

*Міністерство аграрної політики та продовольства України*

*Миколаївський національний аграрний університет*

**кафедра економічної кібернетики і математичного моделювання**

**Веселівська Н.В.**

*Курс лекцій*

*з дисципліни: «МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ НА ПЕОМ»*

*спеціальність 7.10010101 «Енергетика сільськогосподарського  
виробництва»*

*Миколаїв*

*2014*

## **Лекція 1. Основи математичного моделювання.**

### План

1. Поняття «модель» та «моделювання».
2. Етапи математичного моделювання.
3. Основні види та властивості моделей.
4. Принципи моделювання.
5. Пряма та зворотна задачі математичного моделювання.
6. Засоби математичного моделювання.
7. Основні методи розв'язання задач моделювання
8. Математична постановка задачі лінійного програмування.

*В мире не происходит ничего,  
в чем не был бы виден  
смысл какого-нибудь  
максимума или минимума.*  
Л. Эйлер

У сучасній науці істотно зростає роль математики і в розвитку сучасної агрономії. Майбутні спеціалісти потребують серйозної математичної підготовки, яка давала б можливість математичними методами вирішувати широке коло нових проблем, застосовувати обчислювальну техніку, використовувати теоретичні досягнення в практиці. Обробка експериментальних даних з використанням математичної статистики — це лише найбільш поширене, але не єдине і не найважливіше застосування математики. Врахувати взаємодію різноманітних факторів, що визначають структуру і особливості побутування природних систем, можна тільки за допомогою математичних методів і методів математичного моделювання. Важливим слід вважати процес побудови адекватної математичної моделі об'єкта або системи, що вивчається.

Математичне моделювання процесів, явищ та конструкцій складних технічних систем є могутнім інструментом аналізу величезної кількості задач. А імітація на ЕОМ функціонування складних (реально існуючих або проєктованих) систем — практично єдиний спосіб швидкого вирішення актуальних проблем науки та техніки.

Математизація великої кількості прикладних та фундаментальних наук дозволила моделювати поведінку частин системи, що розглядається, їх взаємодію з врахуванням факторів, що впливають на цю систему. Моделювання дозволяє значно спростити планування та виконання експериментів.

### **1. Поняття «модель» та «моделювання».**

Будь-які два об'єкти  $O_1$  та  $O_2$  людської діяльності в чомусь подібні, а в чомусь різні. Заміщення об'єкта  $O_1$  об'єктом  $O_2$  з метою вивчення найважливіших рис  $O_1$  за допомогою об'єкту  $O_2$  називається *моделюванням* об'єкта  $O_1$  об'єктом  $O_2$ . При цьому об'єкт  $O_1$  називають *оригіналом* (або *натурою*), а об'єкт  $O_2$  — *моделлю*.

Таким чином, *модель* – це замісник оригіналу, який дозволяє вивчити або зафіксувати деякі важливі риси оригіналу.

Приклад 1. При проектуванні літаків, ракет, кораблів, автомобілів для дослідження залежності сили опору середовища рухові об'єкта від його швидкості (що практично неможливо визначити аналітично) використовують спосіб продувки зменшеної моделі транспортного засобу в аеродинамічній трубі. Заміщення оригінала моделлю в цьому випадку дозволяє зафіксувати важливі властивості оригінала та дослідити їх.

Приклад 2. При дослідженні роботи електричного генератора на споживача підключити реального споживача до генератора практично неможливо, тому споживача замінюють як правило послідовним підключенням опору та конденсатора. Тут має місце лише фіксація найбільш важливих рис оригіналу.

В загальному випадку процес моделювання складається з таких послідовних етапів:

- 1) Постановка задачі моделювання та визначення рис оригіналу, що підлягають дослідженню;
- 2) Вибір моделі, яка б фіксувала найбільш важливі риси оригіналу та яка б добре піддавалася дослідженню;
- 3) Дослідження моделі згідно поставленої задачі;
- 4) Перенесення результатів дослідження на оригінал;
- 5) Перевірка цих результатів.

Як моделі, що заміщують оригінали, можуть використовуватися зокрема засоби спілкування людей (мова, писемність), засоби пізнання матеріального світу (модель сонячної системи, модель будови атома), засоби навчання та тренування (тренажери для льотчиків, водіїв), засоби прогнозування поведінки оригіналів в різних умовах (наприклад, для підбору параметрів, що відповідають оптимальному функціонуванню роботи приладу в певних умовах).

В практичній діяльності інженера-електрика як в галузі виробництва, так і в галузі проектування або наукової діяльності в основному використовується моделювання з використанням методів та засобів математики, тому подальша увага буде сконцентрована саме на математичному моделюванні.

**Математичне моделювання** – це галузь науки і техніки, що забезпечує виявлення закономірностей проходження різноманітних явищ оточуючого світу або роботи систем та пристроїв шляхом їх математичного опису та моделювання без проведення натурних випробувань. При цьому використовують фундаментальні положення та закони математики, що описують явище, систему чи пристрій, що моделюються, на деякому рівні ідеалізації.

Таким чином, математична модель системи чи пристрою – це їх певний математичний опис, що забезпечує імітацію роботи системи чи пристрою на рівні, достатньо близькому до їх реальної поведінки при натурних випробуваннях.

## 2. Основні види та властивості моделей.

Моделі, що спрощують оригінал, зберігають подібність лише в самому істотному, називаються *гомоморфними*. Подібність в даному випадку однозначна лише в один бік: по моделі неможливо відтворити повністю оригінал, але вона дозволяє дослідити найбільш істотні його властивості. На межі гомоморфізм переходить в *ізоморфізм*. Ізоморфні моделі формально можна поміняти місцями, оскільки між ними існує взаємно однозначна відповідність.

Для розв'язання практичних задач недостатньо подібності. Необхідна можливість експериментувати на моделі. Це досягається перш за все спрощенням системи, обмеженням досліджуваних властивостей.

В залежності від способу відображення властивостей досліджуваної системи через ті чи інші носії всі моделі можна підрозділити на дві великі групи: *матеріальні (фізичні)* і *абстрактні*. По своїй природі фізичні моделі можуть бути механічними, електричними, гідравлічними і т.д. Фізичні моделі будуються на принципах прямої аналогії, коли оригінал і модель можуть відрізнятися лише масштабами, чи коли змінюються носії базових властивостей. Зустрічаються і змішані моделі. Вони припускають реальне втілення тих фізичних властивостей оригіналу, які цікавлять дослідника. Спрощені фізичні моделі, нерідко зменшених габаритів, називаються макетами. Тому фізичне моделювання часто іменують макетуванням. Абстрактні моделі, описують поведінку об'єктів абстрактно-логічними засобами, числовими, знаковими, графічними та інші.

**Математичні моделі.** Вони являють собою формалізовані опису об'єкта або системи за допомогою деякого абстрактного мови, наприклад у вигляді сукупності математичних співвідношень або схеми алгоритму. Розрізняють такі види математичного моделювання: вербальні (словесні), графічні, табличні, аналітичні та алгоритмічні. Нерідко математичні моделі виявляються придатними для опису безлічі систем та явищ в самих різних областях науки, техніки та економіки.

Класифікація математичних моделей багатоаспектна. Її можна здійснювати за різними ознаками і цілями, зокрема: за характером використання початкової інформації, за типом (видом) математичного методу, за мірою конкретизації моделюючого об'єкта тощо. Потрібно також розрізняти математичні моделі за особливостями опису ними просторових характеристик (властивостей) реальної системи.

В залежності від того, як враховуються випадкові фактори моделі поділяють на **детерміновані** (де усі параметри та впливи вважаються причино обумовленими) та **стохастичні**, в яких важливу роль відіграють випадкові змінні (наприклад, в теорії фільтрації білий та гаусовський шуми). Стохастичні математичні моделі описують випадкові процеси, при цьому вважають, що випадковість тих чи інших явищ виражається в термінах ймовірності. З математичної точки зору детерміновані моделі є випадком стохастичних, з ймовірністю здійснення явища одиниця. З класу стохастичних моделей використовують здебільшого регресійні моделі. Вони більш ефективні для вивчення не систем у цілому, а лише ситуацій у системі, де існує (або передбачається) однозначна відповідність між причиною і наслідком. Дуже часто в системах існує функціональний зв'язок між двома і більше змінними. Крім

того, у більшості випадків функціональний зв'язок надзвичайно складний, або навіть зовсім невідомий. Тоді доводиться спиратися на певну гіпотезу щодо характеру функціональної залежності, тобто апроксимувати її деякими відносно простими математичними залежностями. Для пошуку таких залежностей між змінними використовують метод кореляційного та регресійного аналізу. Регресійний аналіз дає змогу побудувати, виходячи з масиву експериментальних даних, рівняння заданого виду. Кореляційний аналіз дає уявлення про те, наскільки тісно експериментальні точки узгоджуються з вибраним рівнянням.

За властивостями в перехідному режимі моделі прийнято поділяти на **статичні** та **динамічні**. В динамічних моделях розглядаються часово-змінні характеристики та взаємодії системи (на відміну від статичних систем, в яких ці параметри не залежать від часу).

В залежності від того, чи є в моделях керовані параметри, вони поділяються на **конструктивні** та **дескриптивні** (описуючі). На відміну від конструктивних описуючі моделі не містять керованих параметрів і тому не дозволяють вирішувати задачу про знаходження найбільш ефективного керування.

До найважливіших типів моделей можна також віднести **матричні** моделі. Назва цієї родини моделей означає, що кількість елементів структури і функцій системи обмежується певною таблицею чисел (матрицею). Моделі цього типу широко використовуються в екології для пошуку простих залежностей при відносно простих розрахунках (найчастіше популяційного характеру). Більш складні варіанти матричних моделей використовують для аналізу таких динамічних явищ, як кругообіг речовин та енергії на рівні балансу (вхід—вихід). Вадодою цього типу моделей є прийняті положення про лінійність зв'язків між досліджуваними елементами природного середовища. Крім того, передбачається сталість структурних коефіцієнтів. В цілому при застосуванні цього типу моделей для екосистеми з'являється потреба у використанні високоагрегованих величин через нестачу багатьох фактичних даних або складність їх отримання. В кінцевому підсумку це може привести до грубих оцінок результатів моделювання.

На відміну від моделей, що характеризують випадкову зміну одного результативного параметра внаслідок дії кількох факторіальних, **багатовимірні** моделі враховують поведінку більш як одного результативного показника. Ці моделі особливо цінні для вивчення складних змінних системи, особливо складних взаємозв'язків між двома множинами. Багатовимірні моделі досить численні і придатні до розв'язування найрізноманітнішою класу задач (описових, оціночних, прогностичних) і не підходять до цілей оптимізації.

За ступенем математичної абстракції детерміновані моделі можна поділити на **складні** та **спрощені структури**. Складні математичні структури, що описують усі причинні зв'язки реальної системи, дозволяють доволі точно спрогнозувати поведінку системи в залежності від зміни параметрів. Спрощені структури, при яких вибирають ряд основних, найсуттєвіших залежностей, устанавлюють і математично описують зв'язки між окремими параметрами, що відповідають причинно-наслідковим закономірностям (ідеалізовані моделі).

Складна система може бути виражена комплексом з детермінованих, стохастичних та ін. моделей. Крім того, реальний процес протікає за обставин, що змінюються. Тому математичні моделі, що адекватно відображають дійсність в

певний момент часу, можуть не відображати умови, що змінилися, в наступний момент. Постає потреба поновлення, адаптації моделей (тобто пристосування до параметрів, які змінилися), що виконується за допомогою ЕОМ.

Крім вищеназваних широких класів математичних моделей, існує ще багато інших класів.

Це, насамперед, **оптимізаційні** моделі. У загальному випадку оптимізаційні задачі можуть бути поділені на три групи: детерміновані, стохастичні та задачі, що вирішуються в умовах невизначеності (або неповної інформації). Останні характеризуються тим, що стохастичні характеристики системи, яка досліджується, не можуть бути отримані. Існуючі методи розв'язування цих задач, як правило, враховують тільки доволі мані зміни коефіцієнтів цільової функції та системи обмежень моделі, і практично не дозволяють врахувати варіації структури моделі. Оптимізаційна задача вимагає переробки великої кількості експериментального матеріалу, його треба проаналізувати та на основі отриманих результатів прийняти рішення для подальшого удосконалення процесу.

Останнім часом успішно намагаються побудувати екологічні моделі, які спираються на теорію катастроф. Моделі цього типу застосовують щодо систем, які мають властивості бімодальності, перервності, гістерезису і дивергенції, а також характеризуються диссипативністю енергетичних та матеріальних потоків. Такими властивостями характеризуються, наприклад, екосистеми ґрунту, їх біогеохімічні цикли та енергетичні потоки. Для екосистеми ґрунту характерним є перебування її у багатьох станах, що пов'язано з добовими, сезонними, річними і багаторічними циклами природної динаміки (бімодальність). Екосистеми ґрунту уповільнено реагують на сторонні впливи (за рахунок потужного механізму гомеостазу)(гістерезис). Ґрунтові системи функціонують так, що близькі вихідні умови нерідко еволюціонують у досить віддалені кінцеві результати (дивергенція). Складніше описати перервність екосистеми ґрунту і процесів, що відбуваються, оскільки ґрунт, як природне тіло, належить до континуальних об'єктів природи.

Широко застосовуються моделі систем, що розвиваються. Їх основною властивістю є така: на момент початку розвитку присутні деякі початкові ресурси. У динамічну систему повинна надходити речовина, енергія та інформація. У системах розвитку повинна бути підсистема відтворення та вдосконалення, має враховуватися характер використання умов довкілля, у взаємодії з яким система створює та поглинає продукти, а також виділяє застарілі та непотрібні. Повинні виконуватися деякі балансові співвідношення між субстратами, що надходять у динамічну систему, та продуктами динамічної системи (причому обов'язковим є функціональний взаємозв'язок між ресурсами, що витрачаються на внутрішній розвиток та на виконання зовнішніх функцій динамічної системи, між швидкістю відтворення ресурсів, інтенсивністю їх використання та результатами функціонування системи).

При програмуванні урожайності сільськогосподарських культур важливе значення має розробка методу багатокритеріальної оптимізації економічних функцій розподілу ресурсів, що базуються на адекватних залежностях „урожайність – комплекс факторів” [1,3], за якими розраховують урожайність при певній величині добрив, води, рівня агротехніки, тощо [2,4].

При програмуванні урожайності необхідно враховувати адекватну виробничу функцію залежності врожаю від різних факторів. При цьому необхідно оптимізувати і іншу функцію - функцію вартості ресурсів, які при цьому витрачаються. Тому в даній роботі ставиться задача двохкритеріальної оптимізації рівнів „вода-добрива” з такого розрахунку, щоб одержувати урожайність при мінімізації вартості ресурсів [3].

Розглянемо самоорганізаційні моделі. На основі принципу самоорганізації моделей на ЕОМ здійснюється перебір моделей різної складності за рядом критеріїв та знаходження оптимальної. За обраними критеріями визначається модель оптимальної структури.

В математичних моделях окремо виділяють **імітаційні** моделі. Термін «імітаційне моделювання» означає, що маємо справу з такими математичними моделями, за допомогою яких результат неможливо попередньо обчислити або передбачити (для передбачення поведінки реальної системи необхідно проводити її експеримент, імітацію, на моделі за заданих початкових даних). Імітація являє собою чисельний метод проведення на ЕОМ експериментів з математичними моделями, які описують поведінку системи впродовж заданого або сформованого періоду часу. Поведінка компонент складної системи та їх взаємодія в імітаційному моделюванні часто описуються набором алгоритмів (реалізуються певною мовою моделювання). Усі ці описи є програмною імітацією, яку необхідно спочатку підладити та випробувати, а потім використати для постановки імітаційного експерименту на ЕОМ. Тому під процесом імітації на ЕОМ розуміють конструювання моделі, її випробування та використання для вивчення певного явища або проблеми. Отже, імітаційні моделі є достатньо складним програмним виробом. Технологія їх проектування — працемісткий процес, що включає методи та засоби, які забезпечують створення та розвиток моделі. Великі розміри системи, складність поведінки і складників, велика вартість розробки вимагають математичного моделювання на всіх етапах проектування такої системи. Тому моделювання супроводжує і процес проектування, і процес випробування, і процес експлуатації складної системи.

### **Основні властивості моделей.**

Моделі мають ряд властивостей, від яких залежить успіх їх застосування в практиці моделювання. Відзначимо лише деякі з них, найбільш важливі:

Адекватність - це ступінь відповідності моделі досліджуваного реальному об'єкту. Вона ніколи не може бути повною. На практиці модель вважають адекватною, якщо вона з задовільною точністю дозволяє досягти цілей дослідження.

Простота (складність) - чим більша кількість властивостей об'єкта описує модель, тим більш складною вона виявляється. Не завжди чим складніше модель,

тим вище її адекватність. Треба прагнути знайти найбільш просту модель, що дозволяє досягти бажаних результатів вивчення.

**Потенційність** (самий корінь) - здатність моделі дати нові знання про досліджуваний об'єкт, спрогнозувати його поведінку або властивості. На основі вивчення математичних моделей, що описують рух планет Сонячної системи з урахуванням закону всесвітнього тяжіння, теоретично були передбачені існування орбіти планет Нептун і Плутон.

#### **4. Принципи моделювання.**

Моделювання базується на декількох основоположних принципах.

**Принцип інформаційної достатності.** При повній відсутності інформації про досліджуваний об'єкт побудова його моделі неможливо. З іншого боку, за наявності повної інформації про об'єкт побудова його моделі не має сенсу. Існує певний рівень апріорної інформації про об'єкт, при досягненні якої може бути побудована його адекватна модель.

**Принцип здійсненності.** Створювана модель повинна забезпечувати досягнення поставленої мети досліджуваного з імовірністю, істотно відрізняється від нуля.

**Принцип множинності моделей.** Даний принцип є ключовим. Мова йде про те, що створювана модель повинна відображати в першу чергу ті властивості реальної системи, які цікавлять дослідника. Відповідно, при використанні будь-якої конкретної моделі пізнаються лише деякі сторони реальності. Для більш повного її дослідження необхідний ряд моделей, що дозволяє з різних сторін і з різним ступенем деталізації розглянути досліджуваний об'єкт.

**Принцип агрегування.** У більшості випадків складну систему можна уявити що складається з агрегатів (підсистем), для адекватного математичного опису яких виявляються придатним деякі стандартні математичні схеми.

**Принцип параметризації.** Цей принцип означає, що модель будується у вигляді відомої системи, параметри якої невідомі.

#### **5. Пряма та зворотна задачі математичного моделювання.**

**Пряма задача:** всі параметри системи, що досліджується, відомі, досліджується поведінка моделі в різних умовах.

**Зворотні задачі:**

- а) **Задача розпізнавання:** визначення параметрів моделі шляхом співставлення даних спостереження та результатів моделювання. За результатами спостережень намагаються визначити, які процеси управляють поведінкою об'єкта, знаходять визначальні параметри моделі. В зворотній задачі розпізнавання треба визначити значення параметрів моделі за відомою поведінкою системи як цілого.

Приклади задач розпізнавання:



- Задача електророзвідки: визначення підземних структур за допомогою вимірювань на поверхні.
  - Задача магнітної дефектоскопії: визначення дефекту в деталі, розташованій між полюсами магніту, за збуреннями магнітного поля на поверхні деталі.
- б) Задача синтезу (задача математичного проектування): побудова математичних моделей систем та пристроїв, які повинні мати задані технічні характеристики. На відміну від задач розпізнавання, що полягають у визначенні параметрів моделі, відповідній реальному стану системи, в задачах синтезу відсутня вимога унікальності рішення. Це дозволяє з кількох можливих рішень обрати найбільш прийнятний результат.

## **6. Засоби математичного моделювання.**

Математичне моделювання суспільних, економічних, біологічних та фізичних явищ, об'єктів, систем та різних пристроїв – важливий засіб пізнання природи та проектування доволі різноманітних систем та пристроїв. Хрестоматійними стали приклади ефективного використання моделювання в створенні ядерних технологій, авіаційних та аерокосмічних систем, в прогнозуванні атмосферних та океанічних явищ, погоди тощо.

Однак для таких сфер моделювання частіше за все потрібні суперкомп'ютери та роки роботи великих груп вчених для підготовки даних для моделювання та його відпрацювання.

Між тим математичне моделювання на рівні рішення більш простих задач, наприклад, в галузі механіки, електротехніки, автоматики, електроніки та радіотехніки (та ще багатьох галузей науки та техніки) як потреб виробництва, так і для потреб проектування чи наукових досліджень в наш час стало доступним багатьом користувачам сучасних ПК. А при використанні узагальнених моделей стає можливим моделювання досить складних систем, наприклад систем управління, електроенергетичних систем або промислових комплексів.

Існує досить багато програмних пакетів, які дозволяють більш легко проводити моделювання різноманітних систем та пристроїв. Для здійснення математичних досліджень та розрахунків зокрема використовуються пакети Mathematica, Maple, Mathcad, для моделювання електронних пристроїв – EWB, для моделювання систем управління та електротехнічних систем – VisSim, для наукових досліджень та для програмування роботи технічних систем – LabView, але найбільше розповсюдження практично у всіх прикладних галузях отримав програмний пакет MATLAB, що має в своєму складі багато спеціалізованих за окремими напрямками чи галузями програмних пакетів. Як складова пакету MATLAB для проведення моделювання різноманітних систем використовується Simulink – система наглядного моделювання, яка і буде використовуватися при проведенні лабораторних робіт.

## **7. Основні методи розв'язання задач моделювання**

На етапі програмної реалізації моделі та реалізації плану експериментів необхідний вибір методів рішення задач моделювання. При цьому використовуються три основні групи методів:

- 1) Графічні - оціночні наближені методи, засновані на побудові та аналізі графіків;
- 2) Аналітичні - рішення, отримані суворо у вигляді аналітичних виразів (Придатні для вузького кола завдань);
- 3) Чисельні - основний інструмент для вирішення складних математичних задач, заснованих на застосуванні різних чисельних методів.

## **ЛІТЕРАТУРА**

1. Математическое моделирование экономических процессов в сельском хозяйстве / Гатаулин А.М., Гаврилов Г.В., Сорокина Т.Н. и др.; Под ред. А.М. Гатаулина. - М.: Агропромиздат, 1990, - 432 с.
2. Доспехов А.М. Методика полевого опыта (с основами статистической обработки результатов исследований). – 5-е изд., доп. и перераб. – М.: Агропромиздат, 1985 - 351 с.
3. Ковальчук П.І., Лазер П.Н., Пендак Н.В., Матяш Т.В., Зябров І.А.. Інтегроване управління водними, земельними та матеріальними ресурсами у зрошуваному землеробстві.// Таврійський науковий вісник: Збірник наукових праць ХДАУ. Вип.52. – Херсон: Айлант.2007.-с.87-92.
4. Вергунова І.М. Основи математичного моделювання для аналізу та прогнозу агрономічних процесів. – К.:Нора-Прінт, 2000. – 146 с.
5. Вергунов В.А., Вергунова И.Н., Шкрабак В.С. Основы математического моделирования для анализа и прогноза агрономических процессов.- Типография С-ПбГАУ/ООО «Литера», 2003.- 219 с.
6. Компьютерное моделирование биотехнологических процессов и систем: Учеб. пособие / Д.С. Дворецкий, С.И. Дворецкий, Е.И. Муратова, А.А. Ермаков. Тамбов: Изд-во Тамб. гос. техн. ун-та, 2005. 80 с.
7. Леснікова І.Ю., Харченко Є.М. Основи роботи і вирішення задач сільського господарства в середовищі електронних таблиць EXCEL. 2002.- 145 с.

## **L2.**

### **ОСНОВИ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ**

Поняття математичної моделі. Основні принципи та прийоми математичного моделювання. Основні види математичних моделей. Технологія моделювання складних систем на ЕОМ. Системний підхід до побудови математичних моделей. Поняття системи. Властивості систем. Постановка задачі математичного моделювання.

«Використання математичних моделей у різних науках, багатьох галузях людської діяльності показало велике значення цього методу для успішного розвитку останніх» (А. М. Тихонов).

Математичне моделювання процесів, явищ та конструкцій складних технічних систем є могутнім інструментом аналізу задач сільського господарства. А імітація функціонування складних (реально існуючих або проєктованих) систем із застосуванням ІКТ — практично єдиний спосіб швидкого вирішення актуальних проблем науки та техніки.

**Математик-соціолог.** Слово «модель» у повсякденному житті ви чули неодноразово. Так, в газетах можна прочитати, наприклад, що запропонована модель гри футбол певної команди виявилася результативною або, навпаки, не виправдала себе. Моделі можна зустріти в хореографії, економіці, інженерії, архітектурі. Моделі літаків, машин, кораблів супроводжують дитинство кожного хлопчика, а моделі одягу мають особливе значення в житті дівчаток.

Зазвичай в різних ситуаціях замість слова модель говорять: «схема», «креслення», «проект» тощо. Однак усі ці поняття мають приблизно один і той самий зміст. Він полягає в тому, що якийсь складне реальне явище ми замінюємо деякою спрощеною схемою, копією. Так, інженер вивчає і передбачає роботу верстата за кресленнями — моделями верстата. Хореограф конструює танець не на сцені, а на папері, де партнери зображаються квадратами, а партнерші — кружечками. Це схема-модель майбутнього танцю. Композитор, створивши мелодію, записує її за допомогою нот. Ноти мелодії — це її модель. Особливого значення набуває моделювання в техніці

*Моделюванням* називають побудову копії (моделі) якогось явища, процесу, об'єкта тощо.

**Інженер-конструктор.** Я хочу сказати, що в науці і практиці добре відомий метод фізичного моделювання. Ним користуються в різних галузях машинобудування, приладобудування, будівельній справі. Адже, перш ніж створити нову машину, літак, корабель чи верстат, інженер-конструктор створює зменшену модель об'єкта, досліджує необхідні параметри його діяльності, вдосконалює, а потім виробляється власне об'єкт — машина, літак та ін. Процеси, які відбуваються в моделі й оригіналі, мають однаковий характер.

Необхідність створення фізичних моделей диктується багатьма практичними завданнями. Наприклад, Волховську ГЕС будували на такому місці, де залягав пісок. За правилами гідротехніки, його потрібно було виймати і греблю будувати на гранітній основі. Все це, звичайно, вимагало багато коштів, матеріалів, часу. Під керівництвом відомого інженера Графтіо була створена модель русла річки, за якою дослідили, як змінюватиме положення гребля під тиском води, якщо вона побудована на піщаній основі. Зроблені висновки мали велике практичне значення. ГЕС було побудовано з найменшими витратами.

**Учений-математик.** Особливу роль у науці та практиці відіграють математичні моделі. Що ж таке математична модель? Розглянемо задачу. Поживність 1 кг сіна складає 0,42, а силосу — 0,20 кормових одиниць. Сіно Містить 85 %, а силос — 27 % сухої речовини. Скільки сіна і скільки силосу потрібно давати корові на добу, якщо вона повинна отримати 6 кормових одиниць і 9 кг сухої речовини?

*Розв'язання.* Нехай раціон корови мстить  $x$  кг сіна і  $y$  кг силосу. Тоді маємо:

$$\begin{cases} 0,4x + 0,2y = 6 \\ 0,85x + 0,27y = 9 \end{cases}$$

$x=3\text{кг}, y=23\text{кг}.$

У цій задачі ми мали справу з нематематичними поняттями: сіно, силос, корова, кормові одиниці. Це прикладна задача. Щоб її розв'язати, складаємо систему рівнянь, що є математичною моделлю даної прикладної задачі.

*Моделлю* називають спеціально створений об'єкт, який відображає властивості досліджуваного об'єкта.

Математичні моделі створюють із математичних понять і відношень: геометричних фігур, чисел, функцій, рівнянь, неровностей, їх систем. Розв'язування прикладної задачі здійснюється в три етапи:

1. Створення математичної моделі запропонованої задачі (для нашого прикладу — це система двох лінійних рівнянь з двома невідомими).
2. Розв'язування відповідної математичної задачі.
3. Аналіз відповіді.

Наприклад, система рівнянь

$$\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2 \end{cases}$$

є математичною моделлю напруги в електричному колі (в електротехніці), рівноваги для системи важелів або пружин (у механіці), завантаженості верстатів (у плануванні) та ін.

Функція  $y = a^x$  є математичною моделлю, яка описує зростання продуктивності праці, зміну атмосферного тиску, розмноження бактерій, приріст деревини, населення, збільшення вантажопідйомності транспорту тощо.

Багато моделей описують об'єкт наближено. Так, закон всесвітнього тяжіння  $F = f \frac{m_1 m_2}{r^2}$  є наближеною моделлю руху планет Сонячної системи. Незважаючи на простоту, модель протягом багатьох років з абсолютною точністю відтворювала особливості руху планет і навіть допомогла теоретично передбачити дві невідомі планети — Нептун і Плутон.

Лікар. Математичному моделюванню відведено значну роль і в медицині. За допомогою математичних моделей можна створити органи людини. Що ж являє собою така модель? У найпростішому випадку — це формули, частіше — системи з десятків і сотень рівнянь, нерівностей, які математично виражають діяльність окремих органів людини, перебіг патологічних процесів в організмі.

Саме з математичним моделюванням, на думку вчених зв'язаний процес визволення людства від багатьох недуг, у тому числі і серцево-судинних захворювань.

А чи можна, не чекаючи спалаху грипу чи іншого інфекційного захворювання, передбачити, коли почнеться епідемія? Так. Учені-медики уже створили математичну модель грипу. Це формули з декількома десятками інтегралів, систем рівнянь, нерівностей. За їх допомогою визначають, коли хвора людина стає особливо небезпечною, яка кількість жителів регіону буде охоплена хворобою і в який період. Такий діагноз дозволяє медикам завчасно підготувати профілактичні та лікувальні засоби.

В Інституті серцево-судинних захворювань створено лабораторію математичного моделювання. Тепер стало можливим створювати індивідуальні математичні моделі відповідно до стану здоров'я кожного пацієнта і за їх допомогою лікувати небезпечні хвороби.

Вчений-еколог. Математичні моделі знаходять широке застосування в екології. Так, у Науково-дослідному інституті водних проблем розроблені математичні моделі прогнозування якості води в Чорному та Азовському морях. Створені моделі дозволяють передбачити негативні зміни, які загрожують екосистемі цих водних басейнів, та вжити своєчасно заходів, що запобігають забрудненню та руйнівній експлуатації їх багатств.

Математики і біологи побудували математичні моделі співіснування видів, один з яких є хижаком відносно іншого. Ця модель є системою двох диференціальних рівнянь. Дослідження цієї моделі показало, що хижак повністю не знищує інший вид. Види співіснують, періодично змінюючи свою чисельність. Тому не можна винищувати, наприклад, вовків. Все в природі має право на життя. Природу треба берегти.

Економіст. Математичні моделі набули широкого застосування і в економіці. Розглянемо задачу. Треба скласти математичну модель заробітної платні робітника згідно з відрядно-преміальною системою.

Нехай  $z$  — заробітна платня,  $x$  — кількість виготовленої продукції,  $a$  — тарифна ставка одиниці продукції,  $p$  — премія за перевиконання норми.

Маємо модель:  $z = ax + p$ .

Економіко-математичні моделі дозволяють розв'язувати багато задач, зокрема так званих оптимізаційних, які виникають у найрізноманітніших видах людської діяльності. Ось одна з таких задач.

У цеху одного із заводів випускають прилади двох видів. Виробнича потужність цеху — 100 приладів першого виду або 300 другого виду. Але відділ технічного контролю в змозі перевірити тільки 150 виробів на добу. Крім того, вироби першого виду вдвічі дорожчі за вироби другого виду. Потрібно скласти такий план випуску продукції, який за даних умов забезпечував би цеху найбільший прибуток.

Учений - кібернетик. Останнім часом у життєвій практиці використовується імітаційне моделювання. Що ж таке імітаційна модель? Це математичка модель з подальшим дослідженням та експериментами на ЕОМ. Імітаційне моделювання застосовується у випадках, коли прямиий

експеримент неможливий. Так, у лабораторії не можна змодельовати атмосферні процеси, зробити термоядерний реактор або штучно відтворити етапи розвитку Всесвіту. Тут на допомогу приходить імітаційне моделювання. Тож математиками, фізиками й біологами були розроблені моделі глобальних біосферних процесів, які можуть відбутися після ядерної війни. Результати імітаційного моделювання показали, що ядерний конфлікт призведе не тільки до загибелі людей від радіації, вибухів, генетичних змін. У результаті ядерних вибухів на всій планеті розпочнуться самопідтримуючі пожежі. В атмосфері будуть носитися величезні хмари сажі, пилу. Настане так звана ядерна ніч. На Землі запанує ядерна зима. Температура знизиться до  $-40^{\circ}$ — $-50^{\circ}$ . Все живе на Землі зникне.

Тож завдання народів світу — зберегти мир на планеті.

Оскільки математичні методи не можуть застосовуватися безпосередньо до досліджуваного об'єкта, необхідною є побудова адекватної цьому об'єкту математичної моделі.

*Математична модель* – наближений опис якого-небудь явища зовнішнього світу, виражене за допомогою математичної символіки.

Поняття математичного моделювання трактується різними авторами по своєму.

Під математичним моделюванням ми будемо розуміти метод дослідження процесів або явищ шляхом побудови їхніх математичних моделей і дослідження цих процесів.

В основу методу покладемо адекватність між змінними складеного рівняння і досліджуваного процесу. Зрозуміло, що на практиці ці процеси не будуть абсолютно ідентичні. Але можна удосконалювати математичну модель, яка більш точно буде описувати цей процес. Треба пам'ятати, що в останньому випадку, як правило, математичні рівняння ускладнюються. А це означає, що їх моделювання на ЕОМ потребує більше часу, або ж більше не визначаючих обчислювальних комплексів.

Схема таких досліджень починається з постановки задачі і закінчується проведенням ефективного обчислювального експерименту. Її умови можна записати в такій формі:

а) математична постановка задачі, при якій різні фізичні, суспільні процеси записуються у вигляді математичних співвідношень;

б) дослідження поставленої задачі. Вибір методу для аналітичного розв'язання.

в) аналіз результатів. Перевірка – чи задовольняє математична модель критерію практики;

г) послідовний аналіз моделі, її уточнення чи побудова нової більш досконалої.

#### **Задача математично програмування**

Нехай  $x_1, x_2, \dots, x_n$  - сукупність керованих змінних,  $q_i(x_1, x_2, \dots, x_n) \leq b_i, i = \overline{1, m}$ , - система нерівностей, які задовольняють умови функціонування складної системи,  $F(x_1, x_2, \dots, x_n)$  – цільова функція. Тоді математичну модель найкращого функціонування системи формулюють як оптимізаційну задачу: знайти найбільше (найменше) значення цільової функції

$$F(x_1, x_2, \dots, x_n) \rightarrow \max(\min) \quad (1)$$

За додаткових обмежень

$$q_i(x_1, x_2, \dots, x_n) \leq b_i, i = \overline{1, m}. \quad (2)$$

Задача знаходження найбільшого або найменшого значення функції  $F$  за умов (2) називається загальною задачею математичного програмування. Сама назва предмета «математичне програмування» пов'язана з тим, що метою розв'язування задач є вибір програми дій.

У структурі моделі (1) можна виділити 3 елементи:

1) Набір керованих змінних  $x_1, x_2, \dots, x_n$ , значення яких підлягають оптимізації. Різні допустимі комбінації значень змінних відповідають можливим розв'язкам задачі.

2) Цільова функція  $F(x_1, x_2, \dots, x_n)$  - функція, що виражає залежність прийнятого критерію оптимальності від керованих змінних.

Критерій оптимальності є мірою наближення розв'язку до поставленої мети. В економічних задачах, як правило, таким критерієм виступає показник ефективності функціонування системи (наприклад, прибуток від реалізації продукції, продуктивність праці, таке інше) або показник витрат. Слід зазначити, що одній меті можуть відповідати декілька критеріїв оптимальності (багатокритеріальна задача); в цьому разі цільова функція має враховувати всі виділені критерії.

3) Умови або обмеження  $q_i(x_1, x_2, \dots, x_n)$ , що накладаються на значення змінних або на співвідношення між ними.

## Коротка класифікація моделей МП

Основними ознаками, за якими моделі математичного програмування поділяють на класи, є: характер функцій у складі моделі, тип змінних, врахування фактору часу та випадкових факторів

В залежності від характеру функцій, що входять до складу моделі, задачі МП можуть бути лінійними або нелінійними. Якщо цільова функція і функції всіх обмежень моделі є лінійними, то дана задача являє собою задачу лінійного програмування (ЗЛП). В інших випадках, якщо хоча б одна функція в складі моделі є нелінійною, маємо справу із задачею нелінійного програмування (ЗНЛП). Зазначимо, що для ЗЛП розроблені універсальний і ціла низка часткових методів розв'язання. Навпаки, лише незначна частина ЗНЛП (а саме, задачі опуклого програмування) може бути ефективно розв'язана частковими методами.

За типом змінних розрізняють задачі МП з неперервними та дискретними змінними. Останні створюють окремий клас задач дискретного програмування, підкласом якого є задачі цілочисельного програмування.

За фактором часу задачі математичного програмування поділяють на статичні та динамічні.

Нарешті, в залежності від того, якими є параметри моделі, - постійними чи імовірнісними величинами, - розрізняють ЗМП детерміновані та стохастичні.

Коротка класифікація моделей математичного програмування представлена на рис. 1.1.

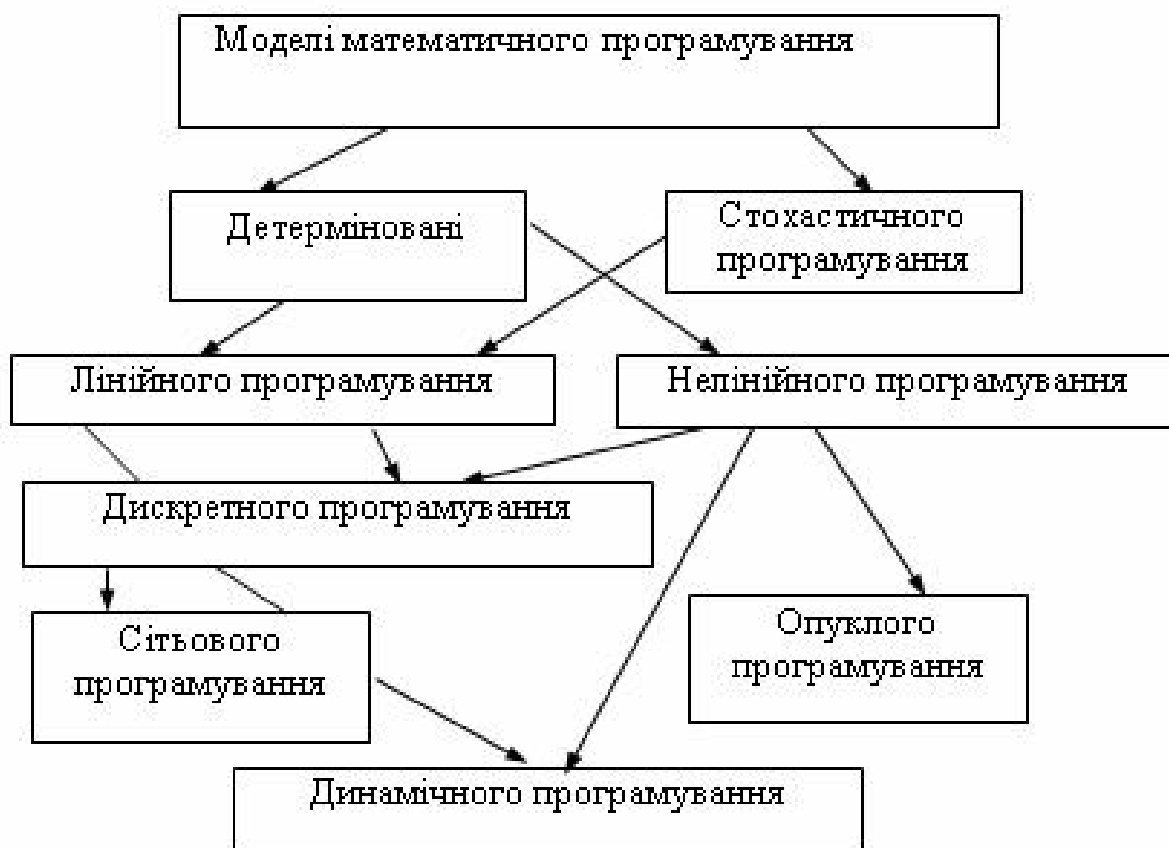


Рис. 1.1. Класифікація моделей математичного програмування.

Системи будь-якої природи, що розвиваються (хімічної, біологічної, технологічної, економічної) мають багато спільного в моделюванні. Математична модель в цьому випадку представляє собою диференціальне рівняння.

## ЛЕКЦІЯ 3.

### Програмування МАТЛАБ (Matlab). Основи

Пакет MatLab був створений компанією Math Works більше десяти років тому. Робота сотень вчених і програмістів спрямована на постійне розширення його можливостей та вдосконалення закладених алгоритмів. В даний час MatLab є потужним і універсальним засобом вирішення завдань, що виникають в різних областях людської діяльності.

Робоча середовище MatLab 6.x, MatLab 7 має зручний інтерфейс для доступу до багатьох допоміжним елементів MatLab.

При запуску MatLab 6.x на екрані з'являється робоча середовище, зображена на рис. 1.

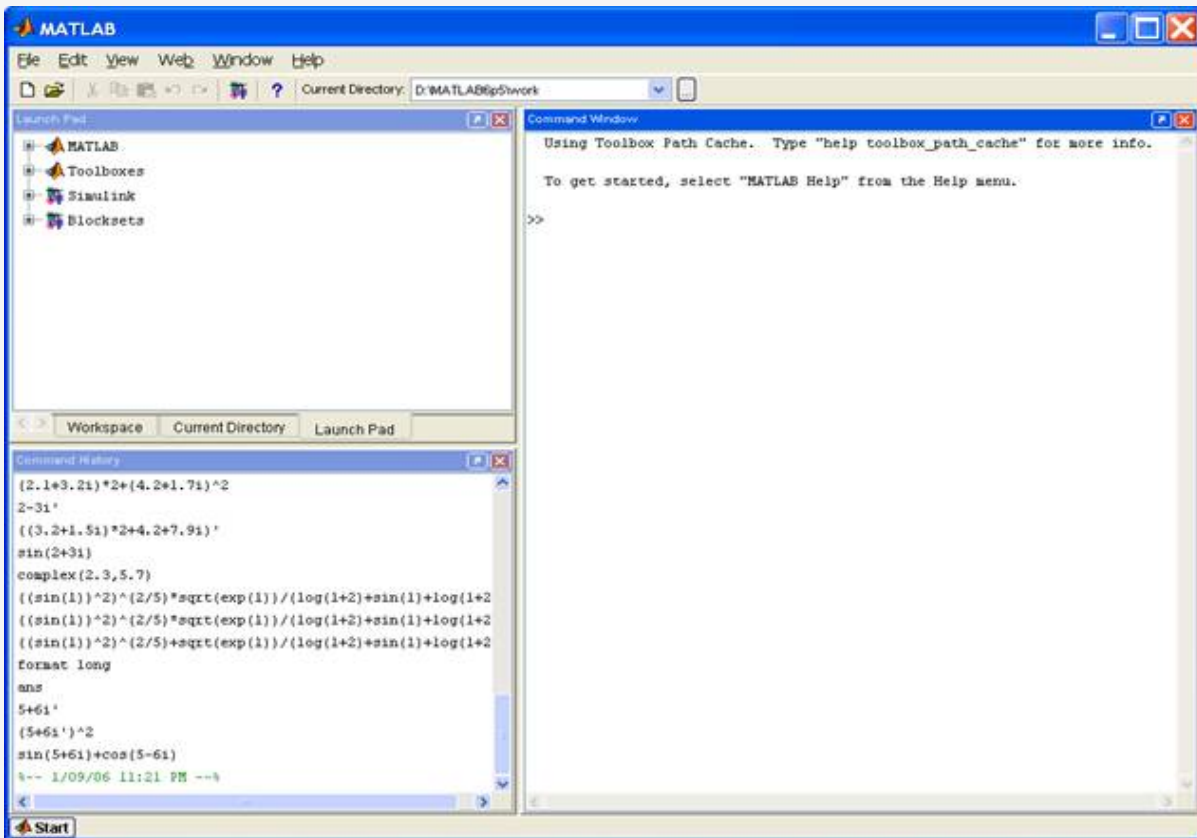


Рис. 1. Робоча середовище пакета MatLab 6.x

Дане заняття вивчає основи роботи (введення) в matlab.

Робоча середовище містить наступні елементи:

- Меню;
- Панель інструментів з кнопками і списком;
- Вікно з вкладками Launch Pad і Workspace, з якого можна отримати простий



доступ до різних модулів ToolBox і до вмісту робочого середовища;

- Вікно з вкладками Command History і Current Directory, призначене для перегляду і повторного виклику раніше введених команд, а також для установки поточного каталогу;
- Командне вікно Command Window з командним рядком, в якій знаходиться миготливий курсор;
- Рядок стану.

Всі команди, описані в цій лабораторній роботі, слід набирати в командному рядку. Сам символ `>`, що позначає запрошення командного рядка, наведений у прикладах, набирати не потрібно. Для перегляду робочої області зручно використовувати смуги скролінгу або клавіші `<Home>`, `<End>` для переміщення вліво або вправо і `<PageUp>`, `<PageDown>` для переміщення вгору або вниз. Про використання клавіш `<up>`, `<down>`, `<right>`, `<left>` буде сказано додатково. Якщо раптом після переміщення по робочій області командного вікна пропала командний рядок з миготливим курсором, просто натисніть `<Enter>`.

Важливо запам'ятати, що набір будь-якої команди або виразу повинен закінчуватися натисканням на клавішу `<Enter>` для того, щоб програма MatLab виконала цю команду або вирахувала вираз.

### Зауваження 1

Якщо в робочому середовищі MatLab 6.x відсутні деякі описані вікна, то слід у меню View вибрати відповідні пункти: Command Window, Command History, Current Directory, Workspace, Launch Pad.

## 2.1. Арифметичні обчислення

Вбудовані математичні функції MatLab дозволяють знаходити значення різних виразів. MatLab надає можливість управління форматом виведення результату. Команди для обчислення виразів мають вигляд, властивий всім мовам програмування високого рівня.

### 2.1.1. Найпростіші обчислення

Наберіть в командному рядку `1 + 2` і натисніть `<Enter>`. У результаті в командному вікні MatLab відображається наступне:

```
»1 +2
```

```
ans =
```

```
3
```

```
»|
```

Що зробила програма MatLab?

Спочатку вона вирахувала суму  $1 + 2$ , потім записала результат в спеціальну змінну **ans** і вивела її значення, рівне 3, в командне вікно. Нижче відповіді розташована командний рядок з миготливим курсором, що позначає, що MatLab готова до подальших обчислень. Можна набирати в командному рядку нові вирази та знаходити їх значення.

Якщо потрібно продовжити роботу з попереднім виразом, наприклад, обчислити  $(1 + 2) / 4.5$ , то найпростіше скористатися вже наявними результатом, який зберігається в змінній **ans**. Наберіть в командному рядку `ans/4.5` (при введенні десяткових дробів використовується крапка) і натисніть `<Enter>`, виходить:

```
»Ans/4.5
```

```
ans =
```

```
0.6667
```

```
»|
```

Зауваження 2

Вид, в якому виводиться результати обчислень, залежить від формату виводу, встановленого в MatLab. Далі пояснено, як задати основні формати виводу.

### 2.1.2. Формати виведення результату обчислень

Потрібний формат виведення результату визначається користувачем з меню MatLab. Виберіть у меню File пункт Preferences. На екрані з'явиться діалогове вікно Preferences. Для установки формату виводу слід переконатися, що в списку лівій панелі обраний пункт Command Window. Завдання формату проводиться з розкритого списку Numeric format панелі Text display.

Розберемо поки тільки найбільш часто використовувані формати. Вибрати short в списку, що розкривається Numeric format в MatLab 6.x. Закрийте діалогове вікно, натиснувши кнопку ОК. Зараз встановлений короткий формат з плаваючою точкою short для виведення результатів обчислень, при якому на екрані відображаються

тільки чотири цифри після десяткової крапки. Наберіть в командному рядку  $100/3$  і натисніть <Enter>.

Результат виводиться у форматі short:

```
»100/3
```

```
ans =
```

```
33.3333
```

Цей формат виводу збережеться для всіх наступних обчислень, якщо тільки не буде встановлений інший формат. Зауважте, що в MatLab можлива ситуація, коли при відображенні занадто великого чи малого числа результат не вкладається у формат short. Обчисліть  $100000/3$ , результат виводиться в експоненційній формі:

```
»100000/3
```

```
ans =
```

```
3.3333e +004
```

Те ж саме відбудеться і при знаходженні  $1/3000$ :

```
»1/3000
```

```
ans =
```

```
3.3333e-004
```

Однак, первісна установка формату зберігається і при подальших обчисленнях, для невеликих чисел висновок результату знову буде відбуватися у форматі short.

У попередньому прикладі пакет MatLab вивів результат обчислень в експоненційній формі. Запис  $3.3333e-004$  позначає  $3.3333 * 10^{-4}$  або  $0.00033333$ .

Аналогічно можна набирати числа у виразах. Наприклад, простіше набрати  $10e9$  або  $1.0e10$ , ніж  $1000000000$ , а результат буде той же самий. Пробіл між цифрами і символом e при введенні не допускається, тому це призведе до повідомлення про помилку:

```
»10 E9
```

```
??? 10 E9
```

Missing operator, comma, or semi-colon.

Якщо потрібно отримати результат обчислень більш точно, то слід вибрати в списку long. Результат буде відображатися в довгому форматі з плаваючою точкою long з чотирнадцятьма цифрами після десяткової крапки. Формати short e і long e

призначені для виведення результату в експоненційної формі з чотирма і п'ятнадцятьма цифрами після десяткової крапки відповідно. Інформацію про формати можна отримати, набравши в командному рядку команду help з аргументом format:

```
»Help format
```

У командному вікні з'являється опис кожного з форматів.

Задавати формат виводу можна безпосередньо з командного рядка за допомогою команди format. Наприклад, для установки **довгого з плаваючою точкою формату** виводу результатів обчислень слід ввести команду format **long e** в командному рядку:

```
»Format long e
```

```
»1.25/3.11
```

```
ans =
```

```
4.019292604501608e-001
```

Зверніть увагу, що команда help format виводить на екран назву форматів прописними літерами. Однак команда, яку треба ввести, складається з малих літер. До цієї особливості вбудованої довідки help треба звикнути. MatLab розрізняє прописні і малі літери. Спроба набору команди прописними літерами приведе до помилки:

```
»FORMAT LONG E
```

```
??? FORMAT LONG.
```

```
[Missing operator, comma, or semi-colon.
```

Для більш зручного сприйняття результату MatLab виводить результат обчислень через рядок після обчислюваного виразу. Однак іноді буває зручно розмістити більше рядків на екрані, для чого слід вибрати перемикач compact (File, Numeric display) з розкритого списку. Додавання порожніх рядків забезпечується вибором loose із списку Numeric display.

### Зауваження 3

Всі проміжні обчислення MatLab виробляє з подвійною точністю, незалежно від того, який формат виводу встановлений.

## 2.2. Використання елементарних функцій

Припустимо, що потрібно обчислити значення наступного виразу:

$e^{-2.5} \times (\ln 11.3)^{0.3} - \sqrt{(\sin 2.45\pi + \cos 3.78\pi) / (\tan 3.3)}$  Введіть у командному рядку це вираз у відповідності з правилами MatLab та натисніть <Enter>:

```
»Exp (-2.5) * log (11.3) ^ 0.3-sqrt ((sin (2.45 * pi) + cos (3.78 pi)) / tan (3.3))
```

Відповідь виводиться в командне вікно:

```
ans =  
-3.2105
```

При введенні вираження використані вбудовані функції MatLab для обчислення експоненти, натурального логарифма, квадратного кореня і тригонометричних функцій. Які вбудовані елементарні функції можна використовувати і як їх викликати? Наберіть в командному рядку команду help eifun, при цьому в командне вікно виводиться список всіх вбудованих елементарних функцій з їх коротким описом. Аргументи функцій полягають в круглі дужки, імена функцій набираються рядковими буквами. Для введення числа л достатньо набрати рі в командному рядку.

Арифметичні операції в MatLab виконуються у звичайному порядку, властивому більшості мов програмування:

- Зведення в ступінь ^;
- Множення і ділення \*, /;
- Додавання і віднімання +, -.

Для зміни порядку виконання арифметичних операторів слід використовувати круглі дужки.

Якщо тепер потрібно обчислити значення виразу, схожого на попереднє, наприклад то необов'язково його знову набирати в командному рядку. Можна скористатися тим, що MatLab запам'ятовує все що вводяться команди. Для повторного занесення їх в командний рядок служать клавіші <up>, <down>.

Обчисліть даний вираз, виконавши наступні кроки.

1. Натисніть клавішу <->, при цьому в командному рядку з'явиться введене раніше

вираз.

2. Внесіть до нього необхідні зміни, замінивши знак мінус на плюс і квадратний корінь на зведення в квадрат (для переміщення по рядку з виразом служать клавіші <right>, <left>, <Home>, <End>).

3. Обчисліть змінене вираз, натиснувши <Enter>.

Виходить

```
»Exp (-2.5) * log (11.3) ^ 0.3 + ((sin (2.45 * pi) + cos (3.78 pi)) / tan (3.3)) ^ 2
```

```
ans =
```

```
121.2446
```

Якщо необхідно отримати більш точний результат, то слід виконати команду `format long e`, потім натискати клавішу <-> до тих пір, поки в командному рядку не з'явиться потрібне вираження, і обчислити його, натиснувши <Enter>.

```
»Format long e
```

```
»Exp (-2.5) * log (11.3) ^ 0.3 + ((sin. (2.45 * pi) + cos (3.78 pi)) / tan (3.3)) ^ 2
```

```
ans =
```

```
1.212446016556763e +002
```

Вивести результат останнього знайденого виразу в іншому форматі можна без повторного обчислення. Слід змінити формат командою `short`, а потім подивитися значення змінної `ans`, набравши її в командному рядку та натиснувши <Enter>:

```
»Format short
```

```
»Ans
```

```
ans =
```

```
121.2446
```

У робочому середовищі MatLab 6.x для виклику раніше введених команд мається зручний засіб - вікно `Command History` з історією команд. Історія команд містить час і дату кожного сеансу роботи з MatLab 6.x. Для активізації вікна `Command History` необхідно вибрати вкладку з однойменною назвою. Поточна команда в вікні зображена на синьому тлі. Якщо клацнути на якій-небудь команді у вікні лівою кнопкою миші, то дана команда стає поточною. Для її виконання в MatLab треба застосувати подвійний клік миші або вибрати рядок з командою за допомогою клавіш <up>, <down> і натиснути клавішу <Enter>. Зайву команду

можна прибрати з вікна. Для цього її треба зробити поточною і видалити за допомогою клавіші <Delete>. Можна виділити кілька йдуть підряд команд за допомогою комбінації клавіш <Shift> + <up>, <Shift> + <down> і виконати їх за допомогою <Enter> або видалити клавіш <Delete>. Виділення послідовно йдуть команд можна виробляти лівою кнопкою миші з одночасним утримуванням клавіші <Shift>. Якщо команди не йдуть одна за одною, то для їх виділення слід використовувати ліву кнопку миші з утримуванням клавіші <Ctrl>.

При клацанні правою кнопкою миші по області вікна Command History з'являється спливаюче меню. Вибір пункту Сміттю призводить до копіювання команди в буфер Windows. За допомогою Evaluate Selection можна виконати зазначену групу команд. Для видалення поточної команди призначений пункт Delete Selection. Для видалення всіх команд до поточної - Delete to Selection, для видалення всіх команд - Delete Entire History.

При обчисленнях можливі деякі виняткові ситуації, наприклад ділення на нуль, які в більшості мов програмування призводять до помилки. При розподілі позитивного числа на нуль в MatLab виходить inf (нескінченність), а при діленні негативного числа на нуль виходить -inf (мінус нескінченність) і видається попередження:

```
»1/0
```

```
Warning: Divide by zero.
```

```
ans =
```

```
Inf
```

При розподілі нуля на нуль виходить NaN (не число) і також видається попередження:

```
»0/0
```

```
Warning: Divide by zero.
```

```
ans =
```

```
NaN
```

При обчисленні, наприклад  $\sqrt{-1}$ , ніякої помилки або попередження не виникає. MatLab автоматично переходить в область комплексних чисел:

```
»Sqrt (-1.0)
```

```
ans =
```

0 + 1.0000i

Як дізнатися, які вбудовані елементарні функції можна використовувати і як їх викликати? Наберіть в командному рядку команду `help eifun`, при цьому в командне вікно виводиться список всіх вбудованих елементарних функцій з їх коротким описом.

## L4-5.

### Призначення системи комп'ютерної математики MATLAB

Комп'ютерна математична система MATLAB (Matrix Laboratory) є універсальним програмним середовищем для виконання науково-дослідницьких та технічних розрахунків практично необмеженої складності. Як мова програмування, MATLAB була розроблена Клівом Моулером у кінці [1970-х](#) років, у той час, коли він був деканом [факультету комп'ютерних наук](#) в [Університеті Нью-Мексико](#). Метою цієї розробки була задача дати студентам факультету можливість використання програмних бібліотек Linpack і Eispack без необхідності вивчення мови програмування FORTRAN. Незабаром нова мова розповсюдилася серед інших університетів і була з великим інтересом прийнята вченими, що працюють в галузі прикладної математики. Дотепер в Інтернеті можна знайти версію [1982](#) року, написану на мові FORTRAN, яка розповсюджена [з відкритим вихідним кодом](#) [4]. У [1983](#) році під час візиту Кліва Моулера в [Стенфордський Університет](#) відбулося його знайомство з інженером Джоном Літлом. Спільними зусиллями вони переписали MATLAB на мову [Cі](#), заснували в [1984](#) компанію The MathWorks для подальшого розвитку цієї програми. Переписані на Cі бібліотеки довгий час були відомі під ім'ям JASCRAS. Спочатку система MATLAB широко використовувалася для викладання лінійної алгебри та чисельних методів, а також для проектування систем управління, але швидко завоювала популярність в багатьох інших наукових та інженерних галузях.

Нині система може використовуватися для розрахунків у багатьох галузях науки і техніки, таких як електро- і радіотехніка, динаміка, акустика, енергетика, економіка та ін. MATLAB є незамінним помічником у виконанні матричного аналізу, під час розв'язування завдань математичної фізики, статистичних,



оптимізаційних і фінансово-економічних завдань, у дослідженні й обробці сигналів і зображень, візуалізації даних, наукової і технічної графіки.

Проте найбільшою мірою система орієнтована на виконання інженерних розрахунків, оскільки її математичний апарат спирається на обчислення з матрицями і комплексними числами. MATLAB містить багато процедур і функцій, необхідних під час виконання складних числових розрахунків і моделювання технічних і фізичних систем.

Основною перевагою системи MATLAB є її відкритість. Іншими словами, процедури і функції MATLAB можна не тільки використовувати, але і модифікувати. Користувач має можливість вводити в систему нові команди або функції, створювати власні програми і процедури або адаптувати існуючі процедури відповідно до своїх потреб. Цьому сприяє проста вбудована мова програмування системи MATLAB. Проте набір вбудованих функцій MATLAB настільки обширний, що для розв'язання більшості завдань важко не знайти відповідну функцію.

Більшість команд і функцій системи зберігаються у вигляді текстових файлів із розширенням .m (так званих m-файлів). Зустрічаються також файли на мові Сі. Усі ці файли, як уже наголошувалося, доступні для редагування. Користувач може створювати як окремі файли, так і цілі бібліотеки файлів, що робить число команд і функцій практично необмеженим, так що MATLAB можна розглядати як майже відкритий програмний продукт. Згадані m-файли можна створювати як за допомогою редактора m-файлів системи MATLAB, так і будь-якого іншого текстового редактора, наприклад в Microsoft Word. Створені користувачем m-файли використовуються так само, як і вбудовані в MATLAB процедури і функції.

Крім роботи з програмами, обчислення за допомогою середовища MATLAB можна виконувати «в режимі калькулятора», тобто отримувати результат відразу ж після введення потрібного оператора або команди.

Графічна галерея системи пропонує величезний набір різних дво- і тривимірних графіків, достатній для візуалізації найрізноманітніших даних. З її допомогою можна малювати лінії, поверхні, стовпчикові, кругові та інші діаграми. Водночас допускається вибір кольору, заливки, способу відображення ліній, поверхонь, масштабу, системи координат і безліч інших параметрів, необхідних для тонкого налаштування графіків.

Зображення можна додатково забезпечувати поясненнями, графічними елементами і навіть складними математичними формулами, а після зберігати їх у внутрішньому форматі (з розширенням .fig) або в популярних графічних форматах.

Система MATLAB містить розроблені пакети розширень з додатковими функціями для їх використання під час розв'язування спеціалізованих завдань. Такі пакети називаються *Toolboxes*. Після встановлення системи MATLAB користувач може вибірково завантажити потрібні йому пакети.

### 3.Склад системи MATLAB

Система MATLAB є об'єднанням математичного ядра, що включає мову програмування високого рівня, бібліотеки стандартних функцій, графічного інтерфейсу користувача і довідкової системи (рис. 1).

*Application Development Tools* – інструментальні засоби розробки додатків, що дозволяють створювати незалежні програмні продукти.

До складу інструментальних засобів входять: **MATLAB Compiler**, **MATLAB Web Server**, **MATLAB Runtime Server**, **MATLAB COM Builder** і **MATLAB Excel Builder**.

За допомогою **MATLAB Compiler**, значно вдосконаленого в останній версії, майже будь-яку (з дуже незначними обмеженнями) програму на мові MATLAB можна на вибір перетворити у виконуваний модуль, бібліотеку **Ci** і **Ci++**, надбудову **Microsoft Excel** і **COM-об'єкт** (для створення двох останніх необхідні модулі **Builder for Excel** і **Builder for COM** відповідно).

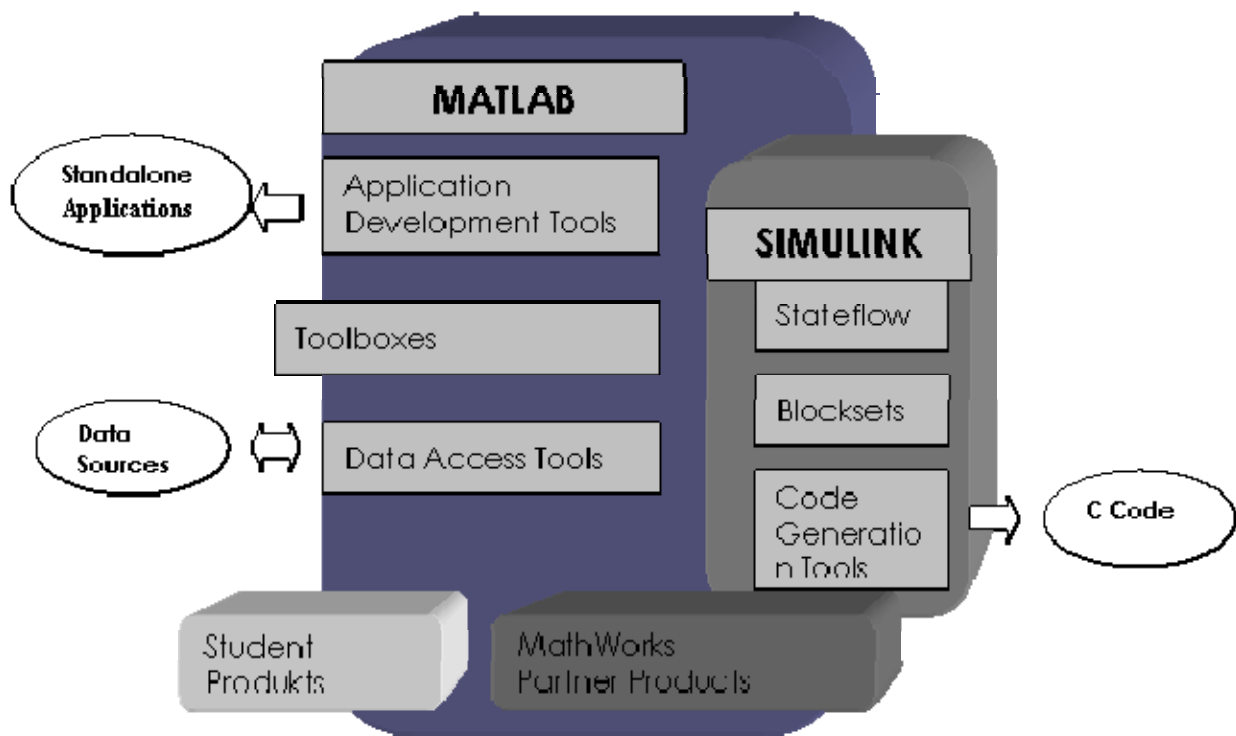


Рис. 1. Склад системи MATLAB

Модуль **MATLAB Web Server** дозволяє користувачам системи MATLAB розробляти додатки для роботи у Web, що використовують стандартні компоненти MATLAB. HTML-документи служать графічним інтерфейсом для розподілених додатків MATLAB. Таким чином, користувачам додатку не обов'язково знати систему MATLAB, більш того, сама система у них може бути навіть не встановлена.

**MATLAB RunTime Server** – це обчислювальний процесор системи MATLAB, поширюваний разом із додатком MATLAB для підтримки його функціонування. Він дозволяє будь-який створений користувачем додаток MATLAB легко і без великих витрат перетворити на незалежний продукт і поширювати, не турбуючись про збереження кодів.

**Toolboxes** – пакети розширення, які призначені для збільшення можливостей систем **MATLAB i SIMULINK** у спеціалізованих галузях: **Математика (Math and Analysis)**, **Імпорт даних (Data acquisition and import)**, **Обробка сигналів і зображень (Signal and Image Processing)**, **Проектування систем управління (Control Design)**, **Фінансові додатки (Financial Modelling and Analysis)**.

**Data Acquisition and Access Tools** – інструментальні засоби обміну даними, що застосовуються для обміну даних із зовнішніми пристроями (осцилографами, генераторами періодичних сигналів та ін.), базами даних (**Oracle, Access**) і іншими додатками.

***MATLAB Student Version*** – студентська версія системи MATLAB, яка дає можливість студентам придбати MATLAB, SIMULINK і пакети символічної математики **Symbolic/ExtendetMath Toolbox** із значними знижками. Інформацію про вартість студентських версій системи MATLAB можна отримати на сайті <http://www.mathworks.com>.

***Simulink*** – інтерактивний інструмент для моделювання, імітації та аналізу динамічних систем. Він дає можливість будувати графічні блок-діаграми, імітувати динамічні системи, досліджувати працездатність систем і вдосконалювати проекти.

***Stateflow*** – середовище розробки і моделювання кінцевих автоматів.

***Blocksets*** – набори спеціалізованих блоків, що розширюють можливості використання стандартної версії **SIMULINK**. Їх використання дозволяє моделювати складні механічні системи, системи гідро-, пневмо- і електроавтоматики.

***Code Generation Tools*** – інструментальні засоби створення коду призначені з метою перетворення блок-схеми моделі, реалізованої в **Simulink**, в оптимальний код на мові програмування Сі для його подальшого запису у вмонтовані додатки.

***Math Works Partner Products*** – пакети розширення, випущені партнерами фірми **MathWorks Inc**. Вони є надбудовою над стандартним набором пакетів розширення, що входять в систему MATLAB.

#### 4.Інтерфейси системи MATLAB

У даний час найбільш популярними є такі версії системи MATLAB: MATLAB 7 (випуск 14), MATLAB 6.5 (випуск 13) і MATLAB 6.0 (випуск 12).

Версія MATLAB 6.5 з'явилась в серпні 2002 року, а її модифікація – MATLAB 6.5.1, яка зазнала з часу виходу лише декілька незначних змін, у вересні 2003 року.

Остання версія системи MATLAB – MATLAB 7 – існує з червня 2004 року.

Версії MATLAB 6\* і MATLAB 7 характеризуються більш зручним інтерфейсом користувача. Новий інтерфейс став багатовіконним: він включає вікно команд (***Command Window***), вікно історії команд (***Command History***), браузер робочої області (***Workspace***) і редактор масивів (***Variable Editor***).

Крім того, в **MATLAB 7** у рядку меню з'явилося дві нових команди – **Desktop**, що дозволяє відображати та приховувати вікна і панелі інструментів робочого столу MATLAB, і **Debug**, елементи якого служать для налагодження програми MATLAB. А ось меню **View**, яке постійно відображалось на екрані у попередніх версіях, тепер з'являється лише тоді, коли є активним вікно **Workspace** чи **Current Directory**.

У більш ранніх версіях MATLAB (до MATLAB 6 ) під час запуску програми відображалось тільки вікно **Command Window**. Оновлений робочий стіл дозволяє легко

управляти вікнами документів, приєднувати графічні вікна до головного вікна програми (щоб вони постійно були видні на екрані), зберігати настройки робочого столу, створювати на панелі кнопки для швидкого виклику команд, що часто використовуються.

Взагалі, вдосконалений інтерфейс MATLAB 7 містить багато нових програмних і налагоджувальних інструментів. З'явилася можливість автоматичного аналізу коду, а також збереження файлів програми безпосередньо в **HTML- і Word-форматах**. Нові інтерактивні інструменти для побудови графіків забезпечують використання їх для швидкої і зручної будови графіків, включаючи можливість генерації коду для повторного багаторазового створення графіків.

В останню версію були внесені й інші суттєві зміни. Зокрема, у версії MATLAB 7 підвищена швидкість і надійність обчислень з числами подвійної точності, а також вдосконалена підтримка інших типів даних – цілочислових та одинарних точностей. Тепер стало можливим і виконання обчислень з цими типами даних. Обробка нових типів даних не вимагає їх переведення в числа подвійної точності, що значно підвищує продуктивність і зменшує обсяг використовуваної пам'яті. Це дозволяє працювати з великими наборами даних. MATLAB тепер використовує бібліотеку цілочислових алгоритмів Intel MMX, що підвищує швидкість обчислень; завдяки новим FFT-алгоритмам швидкість перетворень Фур'є одинарної точності підвищена в середньому на 20%.

Покращений і оновлений компілятор **MATLAB Compiler**, який тепер повністю підтримує мову MATLAB, а також більшість пакетів розширень (**Toolboxes**) системи MATLAB, дозволяє розробникам ефективніше створювати незалежні додатки, або включати їх в **Excel, Cі, Cі++**.

Графічний інтерфейс користувача MATLAB 7, представлений на рис. 2, складається з 4 незалежних вікон, що мають такі назви: **Workspace, Command Window, Command History, Current Directory**.

Вікно *Workspace* використовується для відображення стану зарезервованої області пам'яті комп'ютера, в якій зберігаються значення змінних, використуваних в даному сеансі роботи.

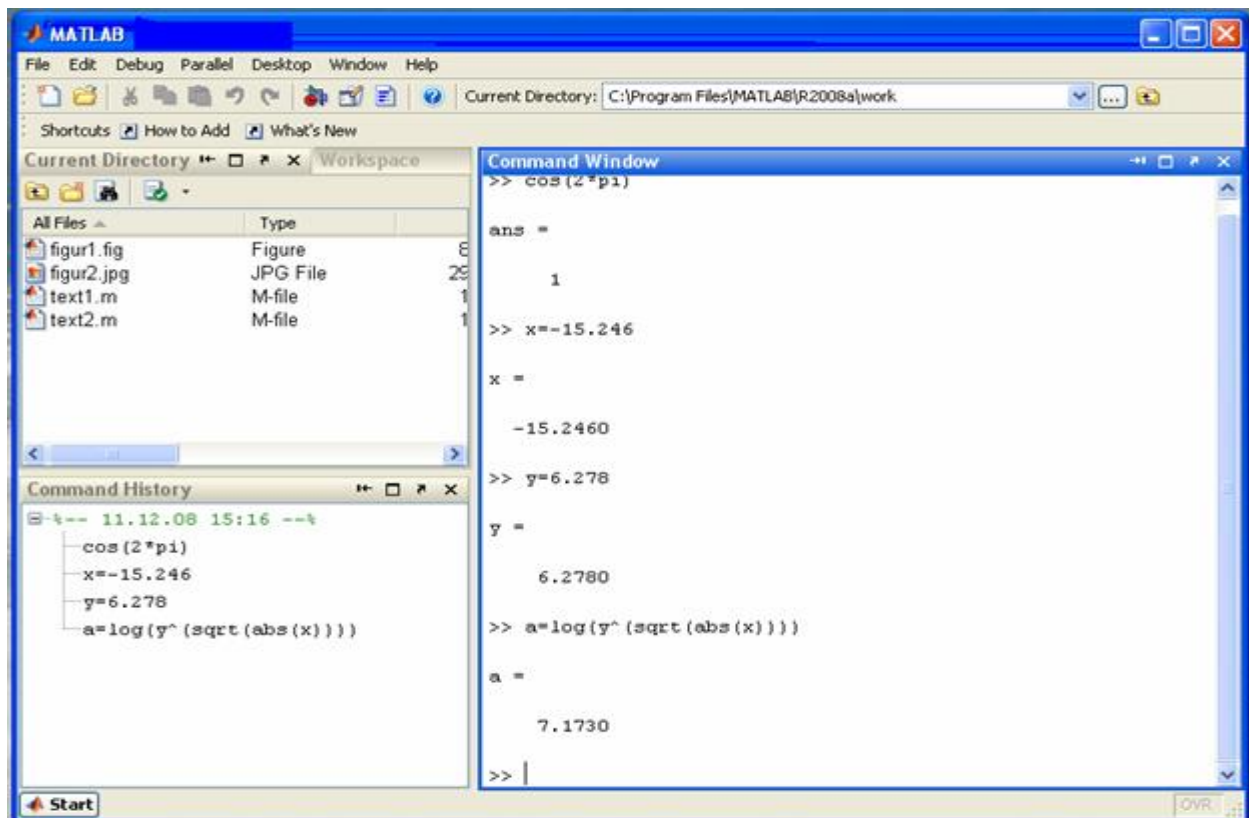


Рис. 2. Графічний інтерфейс користувача

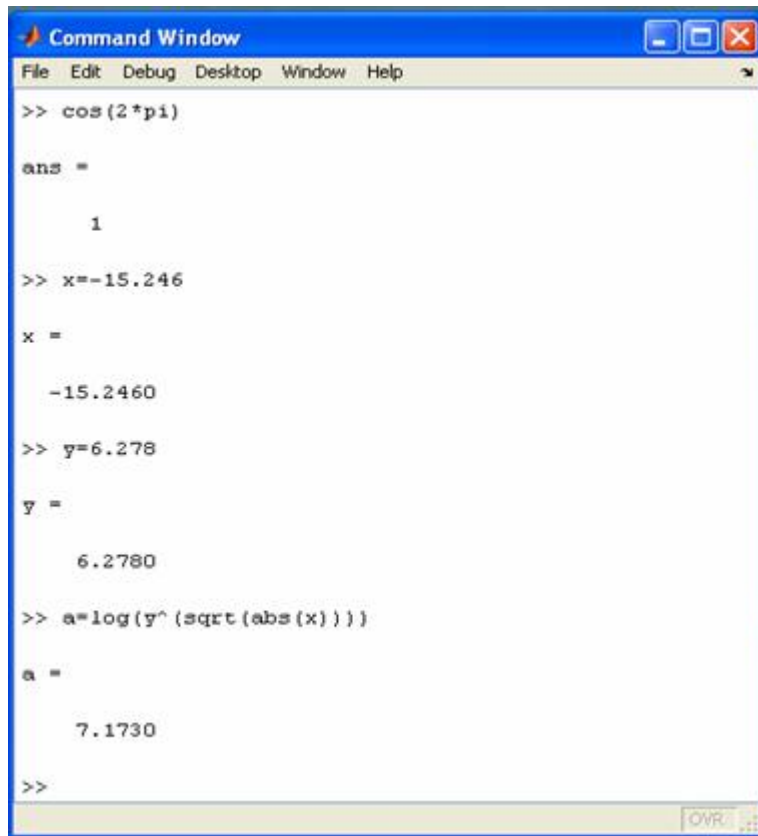
Вікно *Command Window* є основним вікном, в якому вводяться команди і відображаються результати обчислень.

Вікно *Command History* використовується для відображення вмісту буфера, в якому зберігаються виконані раніше команди пакета.

У вікні *Current Directory* відображується список файлів і вкладених папок активного в даний момент каталогу.

Структура вікна **Command Window**, що представлена на рис. 3, аналогічна структурі **Windows-додатків** та складається з рядка меню, робочої області та рядка стану.

У робочій області вікна **Command Window** знаходиться рядок введення команд, відмічений знаком `>>`, в якому можна вводити числа, імена змінних і знаки операцій. Імена змінних повинні починатися з букви і складатися з букв, цифр і розділових знаків. MATLAB розпізнає в іменах змінних до 31 символу, і розрізняє регістр символів. Простими знаками операцій є всім добре відомі знаки арифметичних операцій “+” і “-”. Знак “=” відповідає операції присвоєння. Натиснення клавіші «**Enter**» є для системи MATLAB завданням виконати введену команду і відобразити отриманий результат.



```
Command Window
File Edit Debug Desktop Window Help
>> cos(2*pi)
ans =
    1
>> x=-15.246
x =
   -15.2460
>> y=6.278
y =
    6.2780
>> a=log(y^(sqrt(abs(x))))
a =
    7.1730
>>
```

Рис. 3. Вікно Command Window

Після відображення результату обчислення в командному вікні створюється новий рядок введення команд, відмічений знаком `>>`.

Для перегляду виконаних команд і результатів обчислень, що не вміщуються в командному вікні, є смуги горизонтального і вертикального прокручування. Використання смуг прокручування не відрізняється від інших **Windows-додатків**. Також можна здійснювати прокручування вмісту командного вікна MATLAB за допомогою клавіш клавіатури **PageUp, PageDown, Ctrl+Home і Ctrl+End**.

Необхідно відзначити, що в MATLAB клавіші управління курсором «`↑`» і «`↓`», використовуються інакше, ніж у текстовому редакторі. У MATLAB клавіші «`↑`», «`↓`» використовуються для повернення в рядок введення раніше виконаних команд, кожна з яких перед її виконанням запам'ятовується в стеку команд.

**Стек команд** – це область оперативної пам'яті комп'ютера, відведена для зберігання виконаних в даному сеансі роботи команд. При цьому перегляд стека здійснюється з його кінця, тобто остання виконана команда відобразиться в рядку команд першої. У разі натиснення на клавішу «`↓`» здійснюється прокрутка команд, розташованих у стеку у зворотному напрямі.

Командне вікно MATLAB розділене на дві принципово різних зони: зону перегляду і зону редагування. Виправлення інформації в зоні перегляду неможливе, не дивлячись на те, що в будь-який рядок, що знаходиться в даній зоні можна помістити курсор. Будь-яка спроба редагування тексту, розміщеного в зоні перегляду, приведе до автоматичного переміщення курсору в рядок введення,

розташований в зоні редагування. У зоні перегляду, як і у відомих текстових редакторах, можна за допомогою миші виділити будь-який фрагмент тексту, потім скопіювати його в буфер обміну операційної системи Windows, а потім вставити даний фрагмент в командний рядок MATLAB або документ, створений в будь-якому додатку, що працює під управлінням операційної системи **Windows**.

Зона редагування знаходиться в рядку командного вікна MATLAB, відміченому знаком `>>`. Відзначимо, що є можливість «подовження» командного рядка за рахунок розміщення команди, що вводиться, на декількох фізичних рядках командного вікна. Такий рядок називається логічним рядком введення. Під час введення команди, що розміщується на декількох фізичних рядках, кожен поточний рядок завершується трьома крапками і натисненням на клавішу «**Enter**».

У цьому випадку зона редагування розповсюджується тільки на рядок, у якому знаходиться курсор. Його переміщення на обраний рядок можна здійснювати, як в будь-якому текстовому редакторі, за допомогою миші, після чого можна використовувати клавіші управління курсором «`→`», «`←`». Сумарна довжина логічного рядка введення не може перевищувати 256 символів.

Очищення командного вікна MATLAB здійснюється командою **clc**, яка, проте, залишає незмінним вміст буфера команд і робочої області системи MATLAB.

Значення змінних, обчислених протягом поточного сеансу роботи, зберігаються в спеціально зарезервованій області оперативної пам'яті комп'ютера, яка має назву робочого простору MATLAB (**MATLAB Workspace**).

Для того щоб дізнатися поточне значення будь-якої змінної, розміщеної в робочому просторі MATLAB, досить набрати в командному рядку ім'я змінної і натиснути клавішу «**Enter**». Проте зручнішим, на наш погляд, є використання вікна **Workspace**, в якому відображуються всі змінні, використані в даному сеансі роботи із системою.



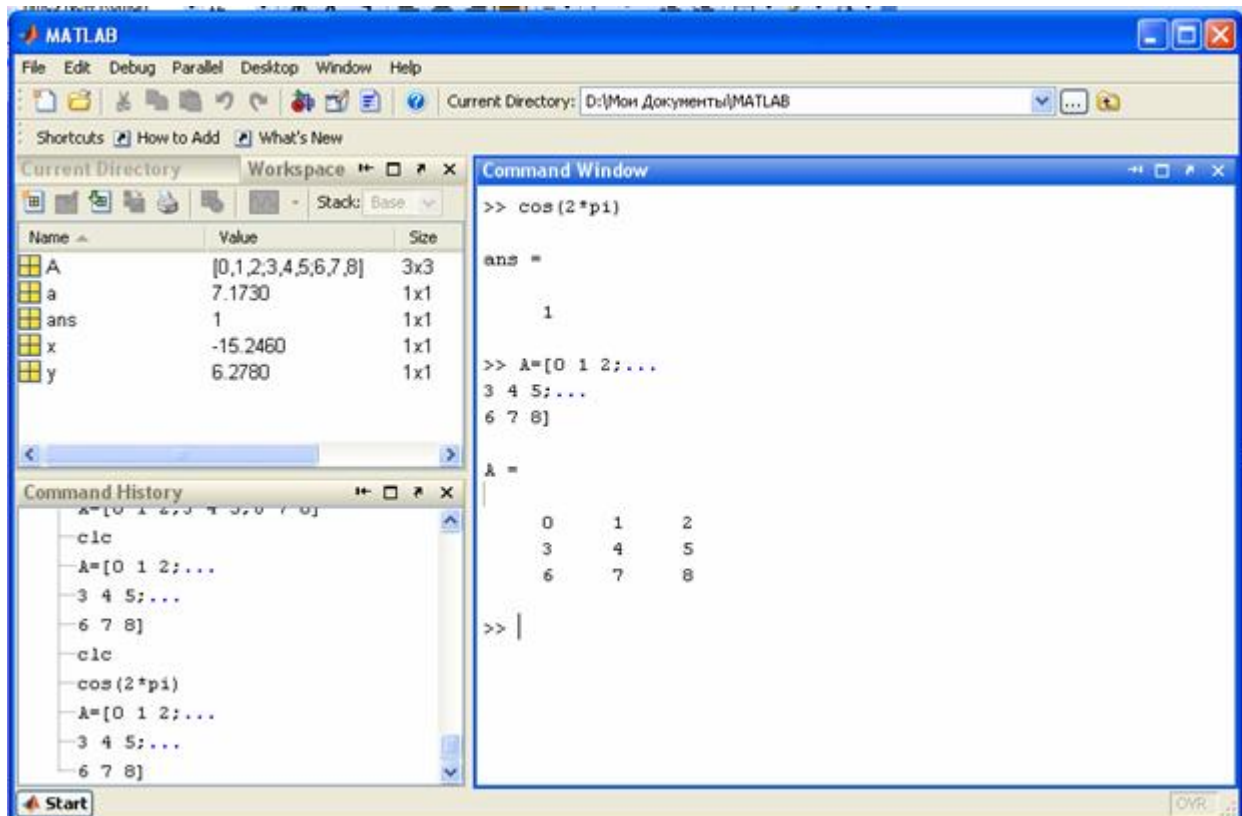


Рис. 4. Робота в командному вікні

Звертання до імені змінної приводить до появи вікна **Variable Editor**, представленому на рис. 5, в якому можна переглядати чи змінювати значення обраної змінної.

Відзначимо, що ефективність роботи системи буде знижуватися у міру збільшення обсягу робочого простору, тому у разі зникнення необхідності зберігання змінних, їх слід видаляти з пам'яті комп'ютера командою, що має такий синтаксис: `clear name 1, name 2 ...`

Для відображення імен змінних, розміщених у даний момент у робочому просторі, потрібно виконати команду **who**.

Дана команда відображає в робочому вікні список усіх змінних, що знаходяться в робочому просторі MATLAB.

Після завершення сеансу роботи із системою MATLAB значення всіх раніше обчислених змінних не зберігаються. Для збереження у файлі на диску комп'ютера вмісту робочого простору, потрібно звернутися до команди меню **File** → **Save Workspace as...**, після чого з'являється стандартне діалогове вікно операційної системи Windows для вибору каталогу на диску та імені файлу. Для файлів, у яких MATLAB зберігає значення змінних із робочого простору, повинно використовуватися розширення `mat`, тому такі файли прийнято називати «MAT-файлами».

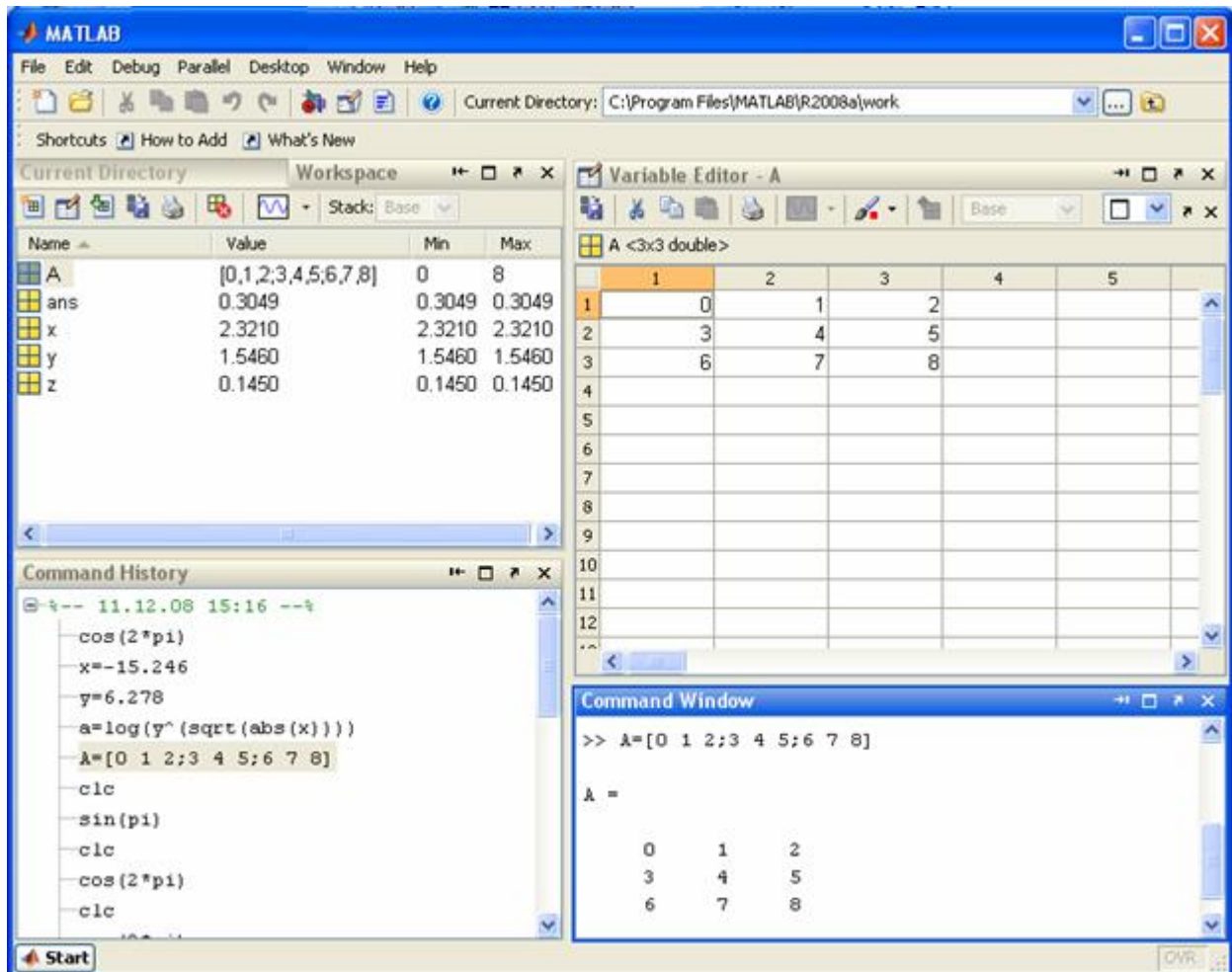


Рис. 5. Вікно Variable Editor

У системі MATLAB до частини команд можна звертатися декількома способами: за допомогою меню головного (командного) вікна, за допомогою кнопок на панелі інструментів і за допомогою введення з клавіатури ключових (зарезервованих) слів із подальшим натисненням клавіші «**Enter**». Наприклад, для збереження вмісту робочого простору замість використання розглянутої команди меню **Save Workspace as** можна безпосередньо в командному вікні MATLAB вказати шлях до папки, в якій буде розміщений даний файл введенням команди: `>> save path'`, шлях до каталогу \ ім'я MAT-файла '. Якщо шлях до файлу, що зберігається, не вказаний, то він буде збережений в поточному робочому каталозі. Відзначимо, що отримати ім'я поточного каталогу можна, виконавши команду **cd**, а змінити поточний каталог введенням команди: `>> cd` шлях до нового каталогу.

У кожному сеансі роботи з MATLAB доцільно як поточний каталог задавати той каталог, з файлами якого належить працювати найчастіше.

Для завантаження в подальших сеансах роботи в оперативну пам'ять комп'ютера раніше збереженого файлу, що містить робочий простір MATLAB, потрібно безпосередньо в командному вікні MATLAB ввести команду: `>> load ім'я MAT-файла.`

Є можливість зчитати із записаного на диску МАТ-файла в робочий простір значення окремих змінних. Для цього використовується попередня команда, доповнена іменами відповідних змінних: `>> load ім'я МАТ-файла, name1`. У результаті її виконання з МАТ-файлу будуть прочитані змінні з ім'ям **name1**.

Про використання будь-якої команди системи MATLAB можна отримати швидку довідку. Для цього у командному необхідно ввести команду `>> help ім'я команди`. Але простіше отримати доступ до внутрішньої впорядкованої довідкової інформації по всіх командах і функціях ядра MATLAB, можна звернувшись до команди меню **Help** → **Product Help**, внаслідок чого з'явиться діалогове вікно, представлене на рис. 6.

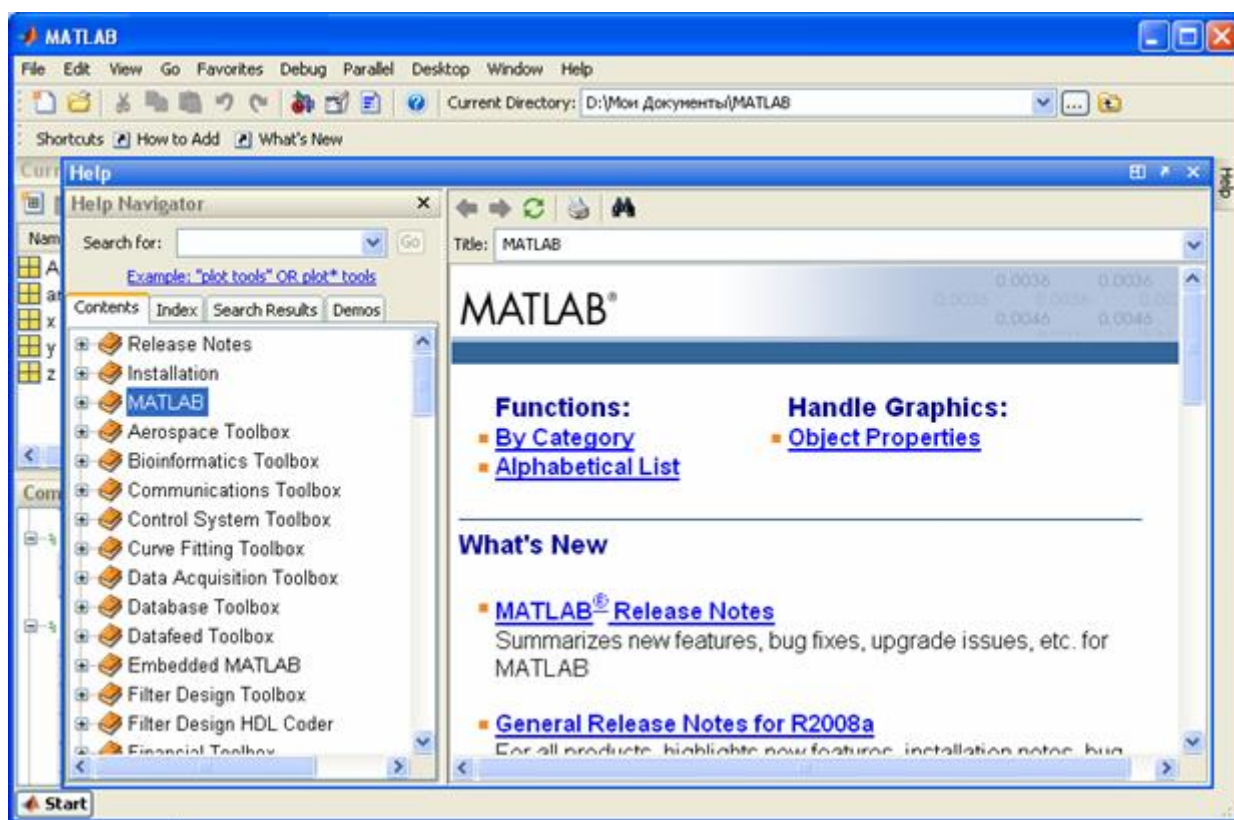


Рис. 6. Вікно довідкової системи

## **Висновок.**

У роботі запропоновано систему комп'ютерної математики MATLAB як середовища для підтримки студентських науково-дослідницьких та технічних розрахунків.

MATLAB є високоефективним засобом для розв'язання широкого спектра обчислювальних завдань і моделювання складних процесів. Водночас пакет достатньо простий, оскільки виконання великого числа операцій здійснюється в звичайному середовищі для користувача, яке потребує лише знань, що відповідають певній дисципліні.

Впровадження системи MATLAB у вищі навчальні заклади I–II рівнів акредитації сприятиме підвищенню якості навчання дисциплін фізико-математичного циклу, розвитку у студентів творчих здібностей, набуванню досвіду використання сучасних інформаційних технологій, що забезпечить успіх у їх професійній діяльності, та знадобиться в подальшому навчанні у вищих навчальних закладах III–IV рівнів акредитації.

## **Список використаної літератури**

1. Закон України «Про вищу освіту» від 17.01.02.
2. Тенденції розвитку коледжів і технікумів
3. *Жалдак М. І.* Комп'ютерно-орієнтовані засоби навчання математики, фізики, інформатики: Посібник для вчителів / М. І. Жалдак, В. В. Лапінський, М. І. Шут. – К. : НПУ імені М.П. Драгоманова, 2004. – 182 с.
4. *Кривилев А. В.* Основы компьютерной математики с использованием системы MATLAB / А. В. Кривилев. – М.: Лекс-Книга, 2005. – 496 с.
5. *Поршнеv С. В.* MATLAB 7. Основы работы и программирования / С. В. Поршнеv. – М.: Бином-Пресс, 2006. – 320 с.

## **L6-8.**

### **Операційна середовище системи Matlab 5**

Операційна середовище системи MATLAB 5 - це безліч інтерфейсів, які підтримують зв'язок системи з зовнішнім світом. Це - діалог з користувачем через командний рядок або графічний інтерфейс, перегляд робочої області та шляхів доступу, редактор і відладчик М-файлів, робота з файлами і оболочкой DOS, експорт та імпорт даних, інтерактивний доступ до довідкової інформації, динамічну взаємодію із зовнішніми системами Microsoft Word, Excel, Microsoft Word, Excel та ін.. Реалізуються ці інтерфейси через командне вікно, інструментальну панель, системи перегляду робочої області та шляхів доступу, редактор/відладчик М-файлів, спеціальні меню і т.п.

#### **1.1. Командне вікно. Інструментальна панель.**

##### **Командне вікно.**

Командне вікно системи MATLAB показано на рис. 1.1. Тут же показано спадаюче меню File.

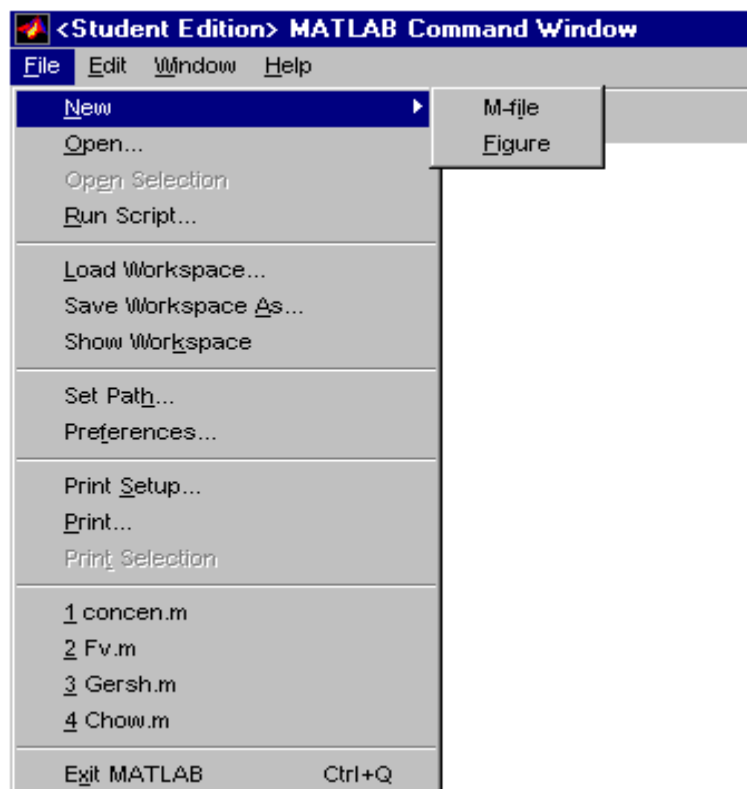


Рис. 1.1

Воно містить наступні опції:

<b>Опції</b>	<b>Підопції</b>	<b>Призначення</b>
New	<a href="#">M-file</a> <a href="#">Figure</a>	Відкрити в редакторі/налагоджувач новий файл. Відкрити графічне вікно
Open		Відкрити в редакторі/налагоджувач вказаний файл
Open Selection		Відкрити в редакторі/налагоджувач вказаний файл, виділений в довільному рядку командного вікна
Run Script		Виклик вікна для запуску Script-файла
Load		Виклик вікна загрузки MAT-файла
Save Workspace As		Виклик вікна збереження MAT-файла
Show Workspace		Виклик засобів перегляду робочої області Workspace Browser
Set Path		Виклик засобів перегляду шляхів доступу

		Path Browser
Preferences		Вибір характеристик
<a href="#">Print Setup</a>		Установка опцій принтера
Print		Установка опцій виводу на друк
Print Selection		Друк виділеного фрагменту

Особливого розгляду заслуговує опція Preferences (Вибір характеристик), яка включає 3 вікна. У першу чергу, розглянемо вікно General (Загальне) (рис. 1.2).

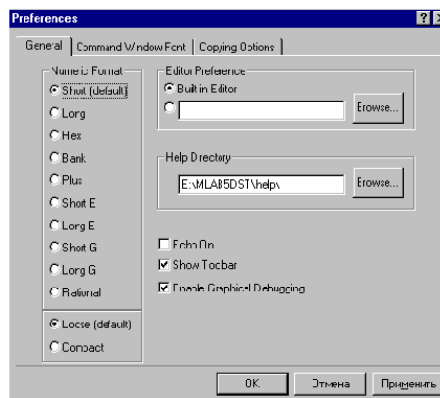


Рис. 1.2

У цьому вікні можна бачити 3 поля і 3 маркера, що мають наступні призначення:

Формат даних	Призначення
<a href="#">Numeric Format</a>	Вибір формату представлення чисел і міжрядкового пробілу. За замовчуванням, формат Short, пробіл Loose
Editor Preference	Вибір текстового редактора. За замовчуванням, вбудований редактор Built in Editor
Help Directory	Каталог довідки Help
Echo on	Показувати на екрані команди виконуваного Script-файлу сценарію / Більше не показувати
Show Toolbar	Показувати на екрані інструментальну панель / Більше не показувати

Enable Graphical Debugging	Підтримувати налагодження графіки / не підтримувати
----------------------------	---

Далі розглянемо вікно CommandWindowFont (Шрифт для командного вікна) (рис. 1.3).

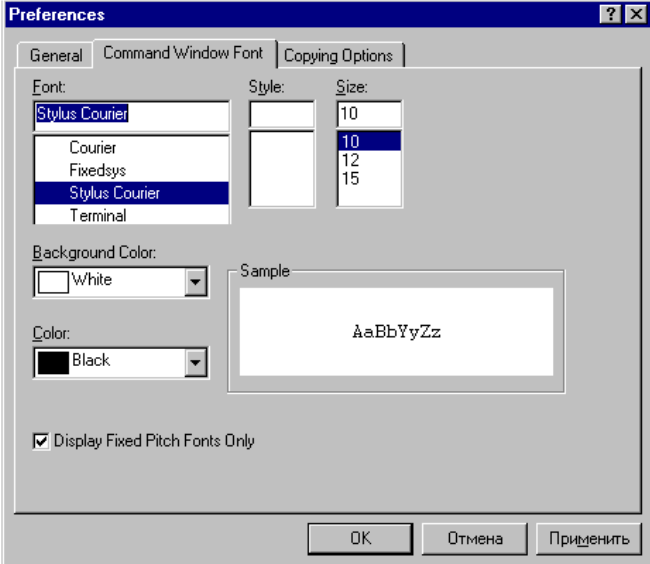


Рис. 1.3

У цьому вікні можна бачити 6 полів і 1 маркер, що мають наступні призначення:

Поле або маркер	Призначення
Font	Шрифт для виводу тексту в командному вікні
Style	Тип шрифту: Light - світлий Regular - нормальний Bold - жирний
Size	Розмір шрифту: 10 12
Background Color	Колір фону: Silver - сріблястий Red - червоний Lime - лимонний Yellow - жовтий Blue - синій Fuscia - світло-фіолетовий Aqua - блакитний White - білий



Color	Колір символів: Black - чорний Maroon - каштановий Green - зелений Olive - оливковий Navy - темно-синій Purple - темно-фіолетовий Teal - зелено-блакитний Gray - сірий
Sample	Зразок фону та шрифту
Display Fixed Pitch Fonts Only	Показати тільки шрифти з фіксованим кроком / Показати всі шрифти

Нарешті, розглянемо вікно CopyingOptions (Опції копіювання) (рис. 1.4).

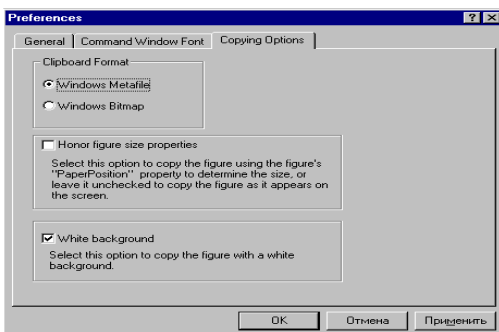


Рис. 1.4

У цьому вікні можна бачити 3 поля, що мають наступні призначення:

Поле або маркер	Призначення
Clipboard Format	Формат копіювання в буфер обміну: Windows Metafile Windows Bitmap

Honor figure size properties	Маркер воспроизведення розмірів малюнка. Вибір цієї опції дозволяє копіювати малюнок з урахуванням властивості "Paper Position" (для формату Windows Bitmap не діє)
White Background	Маркер білого / чорного фону (для формату Windows Bitmap не діє)

### Інструментальна панель.

Інструментальна панель командного вікна системи MATLAB дозволяє забезпечити простий доступ до операцій над М-файлами (рис. 1.5)

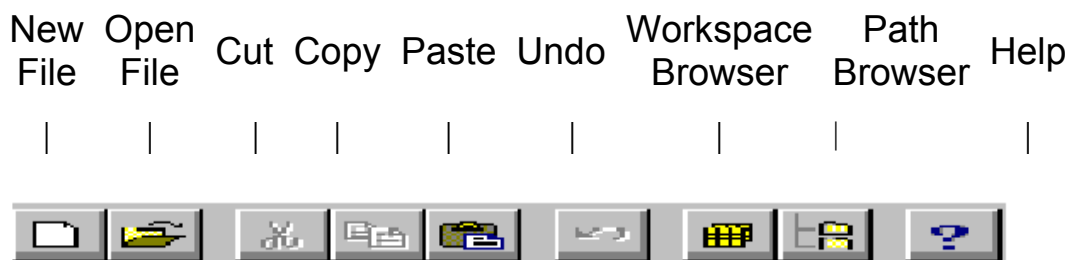


Рис. 1.5

Ці операції включають:

- створення нового М-файлу (New File);
- відкриття існуючого М-файлу (Open File);
- видалення фрагмента (Cut);
- копіювання фрагмента (Copy);
- вставка фрагмента (Paste);
- відновлення тільки виконаної операції (Undo);
- перегляд робочої області (Workspace Browser);
- перегляд шляхів доступу (Path Browser);
- поточна допомога (Help).

## 1.2 Редактор/відладчик М-файлів

До складу системи MATLAB 5 входить редактор/відладчик М-файлів M-file Editor/Debugger, який може бути викликаний з командного рядка командою edit або edit <ім'я М-файлу>.

Інструментальна панель командного вікна редактора/відладчика показана на рис. 1.6.

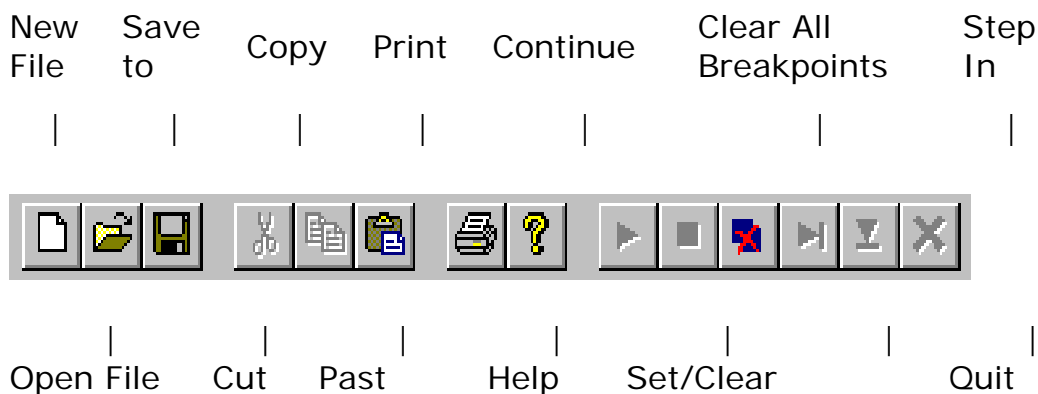


Рис. 1.6

### Редактор/відладчик підтримує наступні операції:

- створення нового М-файлу (New File);
- відкриття існуючого М-файлу (Open File);
- збереження М-файла на диску (Save to Disk);
- видалення фрагмента (Cut);
- копіювання фрагмента (Copy);
- вставка фрагменту (Paste);
- поточна допомога (Help);
- продовжити виконання (Continue);
- встановити/видалити контрольну точку (Set/Clear Breakpoint);
- видалити всі контрольні точки (Clear All Breakpoints);
- виконати один крок налагодження (Single Step);
- увійти в М-модуль (Step In);
- завершити налагодження (Quit Debugging).

### 1.3. Робоча область

Робоча область системи MATLAB - це область пам'яті, в якій розміщені змінні системи. Вміст цієї області можна проглянути з командного рядка за допомогою команд who і whos.

Команда who виводить тільки імена змінних, а команда whos - інформацію про розміри масивів і тип змінної.

Розглянемо як приклад 5 масивів різного типу :

- A - тривимірний масив чисел подвоєної точності;

- B - масив розрідженої структури;
- Z - масив осередків;
- S - масив символів;
- patient - масив записів.

whos

Name	Size	Bytes	Class
A	4x3x2	192	double array
B	4x4	212	sparse array
C	4x3x2	2400	cell array
S	4x16	128	char array
patient	1x2	840	struct array

Загальний підсумок становить 194 елементи, що використовують 3772 байт.

Спеціальний засіб перегляду Workspace Browser забезпечує представлення команди whos у вигляді графічного інтерфейсу. Для того, щоб відкрити Workspace Browser потріб-но або вибрати опцію Show Workspace з меню File menu, або скористатися кнопкою Workspace Browser інструментальної панелі. В результаті цих операцій на термінал буде виведено наступне вікно(мал. 1.7)

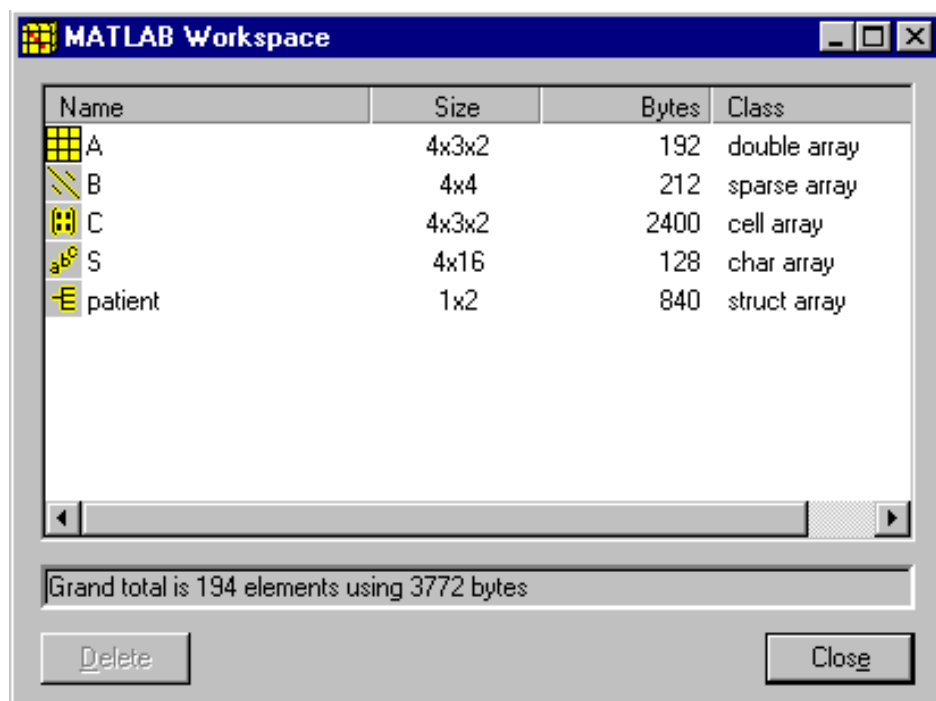


Рис. 1.7.

У цьому вікні можна виконати наступні операції:

- видалити змінну, якщо виділити її і натиснути кнопку Delete;
- закрити вікно за допомогою кнопки Close.

Крім того, можна змінювати розміри колонок за допомогою переміщення їх меж за допомогою миші. Можна виконати перейменування змінної, якщо спочатку виділити її, потім одноразово клацнути лівою клавiшею миші (помітимо, що подвійне клацання ніякої дії не чинить). Після короткої затримки з'являється поле, в якому можна вказати нове ім'я; і нарешті, слід натиснути клавiшу Enter, щоб підтвердити завершення операції.

### **Завантаження і збереження робочої області.**

Команди **save** і **load** дозволяють у будь-який момент часу зберегти вміст робочої області або завантажити нові дані в процесі сеансу роботи. За допомогою цих команд можна також здійснювати експорт і імпорт ASCII- файлів.

#### **Збереження змінних робочої області.**

Команда **save** дозволяє зберегти вміст робочої області в двійковому MAT- файлі, який можна в подальшому викликати командою **load**. Команда **save** також доступна в якості опції **Save Workspace** меню **File**.

#### **Специфікація формату файлу.**

Для того, щоб управляти форматами файлів, слідує в команді **save** на додаток до імені файлу і списку змінних використати наступні прапори:

<b>Прапор</b>	<b>Пояснення</b>
<b>-mat</b>	Двійковий MAT- файл(за умовчанням)
<b>-ascii</b>	ASCII-формат (8 цифр)
<b>-ascii -double</b>	ASCII-формат (16 цифр)
<b>-ascii -double -tabs</b>	Формат з роздільниками і мітками табуляції
<b>-v4</b>	Формат версії MATLAB 4
<b>-append</b>	Додати дані до існуючого MAT- файлу

При використанні прапора **v4** можна зберегти тільки ті дані, які сумісні з даними, використовуваними у версії MATLAB 4; це означає, що зберегти такі типи даних як масиви записів, осередків, багатовимірні масиви або об'єкти не можна. Коли вміст робочої області зберігається в ASCII- форматі, то рекомендується одноразово зберігати тільки одну змінну. Якщо зберігається більше за одну змінну, то система MATLAB створить файл ASCII- файл, який не можна буде надалі завантажити в MATLAB, використовуючи команду **load**.

#### **Завантаження робочої області.**

Команда **load** дозволяє завантажити MAT- файл, який був раніше збережений за допомогою команди **save**. При завантаженні MAT- файлу нові значення однойменних змінних будуть записані замість старих.

Якщо MAT- файл має розширення, що відрізняється від .mat, то необхідно використати прапор - mat; інакше MATLAB вважатиме форматом файлу ASCII-формат.

### Завантаження файлів даних в ASCII- форматі.

Команда **load** дозволяє виконувати імпорт файлів даних в ASCII- форматі; вона перетворить вміст файлу в змінну з ім'ям файлу тільки без розширення.

Наприклад, застосування команди `load tides.dat` створює в робочій області системи MATLAB змінну з іменем `tides`. Якщо початковий файл в ASCII- форматі має `m` рядків з `n` значеннями в кожному рядку, то результатом буде масив чисел розміру `m×n`.

### Використання імен у форматі рядків.

Якщо імена файлів і змінних є строковими змінними, то можна, використовуючи властивість дуальності команди і функції, розглядати команди `load` і `save` як функції. В цьому випадку вхідні змінні повинні слідувати в тому ж порядку, як вони слідували в командному рядку.

Наприклад, послідовність операторів

```
save('myfile', 'VAR1', 'VAR2')
```

```
A = 'myfile';
```

```
load(A)
```

це те ж саме, що і послідовність команд

```
save myfile VAR1 VAR2
```

```
load myfile
```

Для збереження або завантаження послідовності файлів, імена яких мають загальний корінь і додатковий цілочисельний суфікс, необхідно використати структуру циклу.

Наприклад, наступна конструкція дозволяє зберегти квадрати чисел від 1 до 10 у файлах з іменами `data1`, ..., `data10`:

```
file = 'data';
```

```
for i = 1 : 10
```

```
    j = i.^2;
```

```
    save([file int2str(i)], 'j');
```

```
end
```

### Використання групового символу.

Команди **load** і **save** допускають використання групового символу(\*) в якості заміни ряду символів в шаблоні імені змінної.

Наприклад, команда `save rundata x*` зберігає усі змінні, імена яких розпочинаються з символу `x` у файлі з ім'ям `rundata.mat`.

Точно також команда `load testdata ex1*95` завантажує усі змінні, імена яких розпочинаються з символів 'ex1' і закінчуються символами '95', незалежно від того, які символи розміщені між ними.

#### 1.4. Список шляхів доступу

Для пошуку М-файлів система MATLAB використовує механізм шляхів доступу, оскільки М-файли записуються в каталоги або теки файлової системи.

Наприклад, при пошуку файлу з ім'ям `foo` MATLAB виконує наступні дії:

- переглядає, чи не є `foo` ім'ям змінної;
- переглядає, чи не є `foo` вбудованою функцією;
- шукає в поточному каталозі М-файл з ім'ям `foo.m`;
- шукає М-файл з ім'ям `foo.m` в усіх каталогах списку шляхів доступу.

Реально вживані правила пошуку являються більш складними із-за обмежень, які пов'язані з використанням підфункцій, приватних функцій і об'єктно-орієнтованих механізмів. Проте приведений вище порядок пошуку точно відбиває механізм пошуку М-файлів, з якими зазвичай працює користувач.

#### Робота зі списком шляхів доступу.

В процесі сеансу роботи можна вивести на термінал або внести зміни в список шляхів доступу, використовуючи наступні функції:

`path` виводить на екран список шляхів доступу;

- `path(s)` замінює існуючий список списком `s`;
- `addpath /home/lib` і `path(path, '/home/lib')` -- додають новий каталог в список

шляхів доступу;

- `rmpath /home/lib`-- видаляє шлях `/home/lib` зі списку.

Список шляхів доступу, використовуваний за умовчанням, визначений у файлі `pathdef.m`, який розміщений в каталозі `local`; цей файл виконується при кожному запуску системи MATLAB.

Окрім роботи з командного рядка існує засіб перегляду шляхів доступу `Path Browser`, яке підтримує зручний графічний інтерфейс для перегляду і зміни списку шляхів. Проте переважніше вносити безпосередні зміни в М-файл `pathdef.m`, використовуючи який-небудь текстовий редактор, у тому числі і редактор/відладчик системи MATLAB.

#### Поточний каталог.

Система MATLAB використовує поняття поточного каталогу при роботі з М- і МАТ-файлами під час сеансу роботи. Початковий поточний каталог визначений у файлі запуску, який асоційований з ярликом запуску системи MATLAB, розташованому на робочому столі. Клацання правої кнопки миші, встановленої на цьому ярлику, і вибір опції **Properties** дозволяє змінити початковий каталог, використовуваний за умовчанням.

Для виведення поточного каталогу на екран терміналу призначена команда `cd`. Для зміни поточного каталогу слід використати команду

**cd <новий шлях доступу>.**

### Перегляд списку файлів.

Ми вже бачили, як команда `path` дозволяє відобразити список шляхів доступу. У свою чергу, команда `what` дозволяє побачити список файлів, розташованих в заданому або поточному каталогах. Команда `what` без параметрів виводить на екран список файлів поточного каталогу, а команда `what <повний або частковий шлях доступу>` виводить на екран список файлів заданого каталогу.

Для роздруку утримуваного М-файлу призначена команда

`type <ім'я файлу>;`

для редагування М-файлу використовується команда

`edit <ім'я файлу>.`

### Засіб перегляду шляхів доступу.

На платформі PC є засіб візуального перегляду шляхів доступу Path Browser, яке дозволяє переглядати, модифікувати шляхи доступу і бачити списки усіх файлів системи MATLAB.

Для того, щоб відкрити засіб перегляду Path Browser слід використати або опцію Set Path з меню File, або кнопку інструментальної панелі Path Browser.

Вікно засобу перегляду шляхів доступу Path Browser показано на рис. 1.8.

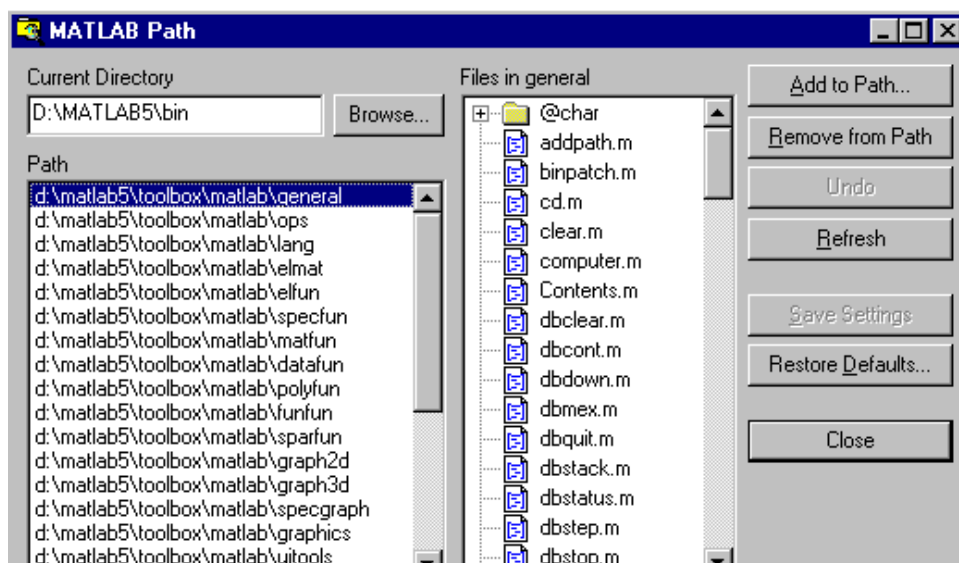


Рис. 1.8

У цьому вікні є:

- поле Current Directory з кнопкою Browse, призначене для зміни поточного каталогу;
- поле Path містить список шляхів доступу;
- поле Files in <ім'я каталогу>, виділеного в полі Path містить список файлів і внутрішніх каталогів типу private @;



- КНОПКИ:

<b>Add to Path</b>	Додати каталог в початок шляху
<b>Remove from Path</b>	Видалити каталог з шляху
<b>Undo</b>	Відмінити попередню операцію
<b>Refresh</b>	Оновити, використовуючи поточні установки
<b>Save Settings</b>	Зберегти установки у файлі pathdef.m
<b>Restore Defaults</b>	Відновити установки, прийняті за умовчанням
<b>Close</b>	Закрити Path Browser

Для переміщення каталогу в іншу позицію в Path слід захопити його лівою кнопкою миші і перемістити в потрібну позицію.

Якщо зміна списку шляхів доступу виконується в командному вікні, то для відображення цих змін в засобі перегляду Path Browser необхідно використати кнопку Refresh.

Усі зміни, які вносяться в список шляхів доступу, діють тільки впродовж сеансу роботи; для того, щоб внести їх у файл pathdef.m для постійного використання необхідно скористатися кнопкою Save Settings.

### 1.5. Робота з файлами і оболонкою DOS. Імпорт і експорт даних

Команди **cd**, **dir**, **delete**, **type** дозволяють з командного рядка системи MATLAB виконати ряд команд DOS, пов'язаних з управлінням файлами. Приведена таблиця відбиває зв'язок команд системи MATLAB з командами DOS :

<b>MATLAB</b>	<b>MS-DOS</b>
cd	chdir
dir	dir
delete	del или erase
type	type

Більшість цих команд дозволяють вказувати шляхи доступу, імена дисководів, використати групові символи.

Запуск зовнішніх програм. Ознакою переходу до виконання команд DOS є знак "!", який вказує, що команда, що йде за ним, - це команда DOS. Це винятково корисно при виклику утиліт і виконанні зовнішніх інших програм без виходу з системи MATLAB.

### Імпорт і експорт даних

Існує багато прийомів для переміщення даних між системою MATLAB і іншими застосуваннями. У більшості випадків при роботі з даними системи MATLAB можна просто використати команди читання і запису файлів. Для складніших наборів даних можна створити власні програми для читання і запису на мовах C або Fortran. Імпортування даних в систему MATLAB. Існує декілька способів для передачі даних з інших застосувань в систему MATLAB. Вибір способу залежить від об'єму і формату даних.

Введення даних у вигляді списку. Якщо кількість даних невелика, то їх можна просто надрукувати, поміщаючи в квадратні дужки. Цей метод незручний при великій кількості даних, оскільки їх неможливо редагувати.

Формування даних в М-файлі. Використовуючи текстовий редактор, можна сформувати М-файл, в якому дані представлені як список елементів, це той же перший спосіб, але він має ту перевагу, що дозволяє за допомогою редактора коригувати дані. Досить після виправлення перезапустити М-файл, щоб ввести виправлені дані.

Завантаження даних з ASCII- файлу. ASCII- файли накопичують дані в 7-розрядному коді без контролю по парності. Кожен рядок містить однакову кількість значень, розділених пропусками, і завершується символом повернення каретки. Ці файли можна редагувати, використовуючи звичайний текстовий редактор. Їх можна читати безпосередньо в системі MATLAB, використовуючи функцію load. При цьому створюється змінна, ім'я якої співпадає з ім'ям файлу. Можна скористатися функцій dlmread, щоб вказати інший тип роздільника.

Читання даних з використанням функцій введення/виведення. Застосування функцій введення/виведення, а також функцій fopen і fread, корисно при завантаженні файлів даних з інших застосувань, що використовують спеціальні формати даних.

Використання спеціальних засобів для читання файлів. Для читання файлів, записаних в спеціальних форматах, в системі MATLAB є наступні спеціалізовані функції:

<b>Функція</b>	<b>Призначення</b>
<b>dlmread</b>	Читання ASCII- файлів
<b>wk1read</b>	Читання електронних таблиць у форматі WK1
<b>imread</b>	Читання зображення з графічного файлу
<b>auread</b>	Читання звукового файлу з розширенням .au(формат фірми SUN Microsystems)
<b>wavread</b>	Читання звукового файлу з розширенням .wav(формат фірми Microsoft)

Створення MEX- файлу. Найкращий спосіб створення програм для побудови даних - це використати вже наявні програми на мовах C або Fortran для читання даних з інших застосувань. Проте цей метод, що називається змішаним

програмуванням, вимагає написання спеціальних програм-зв'язок, що оформляються у вигляді MEX- файлів.

Розробка програми на мовах Fortran або C. Програмісти, що використовують мови Fortran або C, можуть написати спеціальні програми для перетворення даних у формат MAT- файлу системи MATLAB.

В цьому випадку перетворені дані можуть бути завантажені в систему MATLAB за допомогою звичайної команди **load**.

### **Експортування даних з системи MATLAB.**

Існує декілька способів для передачі даних з системи MATLAB в інші застосування.

Використання команди `diary`. Для масивів невеликих розмірів можна використати команду `diary`, щоб створити файл щоденника, який включає команди MATLAB, використовувані впродовж сеансу роботи, а також дозволяє на екрані проглянути необхідні дані. Записи щоденника можуть бути корисні для вкладення в документи або звіти. Надалі можна використати текстовий редактор для редагування щоденника.

Збереження даних у форматі ASCII. Команда `save` з опцією `-ascii` дозволяє записати дані в цьому форматі, причому, використовуючи команду `dlmwrite`, можна задати інший тип роздільника.

Використання спеціальних засобів для запису файлів. Для запису файлів в спеціальних форматах, визначуваних додатками, в системі MATLAB є наступні спеціалізовані функції:

<b>Функція</b>	<b>Призначення</b>
<b>dlmwrite</b>	Запис даних в ASCII- файл
<b>wk1write</b>	Запис даних в електронну таблицю у форматі WK1
<b>imwrite</b>	Запис зображення в графічний файл
<b>auwrite</b>	Запис даних в звуковий файл з розширенням .au(формат фірми SUN Microsystems)
<b>wavwrite</b>	Запис даних в звуковий файл з розширенням .wav(формат фірми Microsoft)

### **Створення MEX- файлу.**

Найкращий спосіб створення програм для запису даних - це використати вже наявні програми на мовах C або Fortran для запису даних в інші додатки. Проте цей метод, ще називається змішаним програмуванням, вимагає написання спеціальних програм-зв'язок, що оформляються у вигляді MEX- файлів.

### **Розробка програми на мовах Fortran або C.**

Програмісти, що використовують мови Fortran або C, можуть написати спеціальні програми для перетворення даних з формату MAT- файлу системи

MATLAB у формат додатка. В цьому випадку дані можуть бути вивантажені з системи MATLAB за допомогою звичайної команди save.

### Текстові файли з роздільниками.

Функції **dlmread** і **dlmwrite** дозволяють читати і записувати дані, відокремлені роздільником, використовуючи ASCII- файл. В якості роздільника може бути використаний будь-який символ, який відділяє одно значення від іншого.

**Наприклад**, розглянемо файл з ім'ям ph.dat, який містить дані, розділені крапкою з комою :

```
7.2; 8.5; 6.2; 6.6  
5.4;9.2;8.1;7.2
```

Для того, щоб прочитати вміст цього файлу в масив з ім'ям A, потрібно використати наступний оператор

```
A = dlmread('ph.dat', ',');
```

Другий аргумент функції **dlmread** вказує тип роздільника.

На додаток до роздільника, який ви використовуєте, функція **dlmread** також вважає роздільниками наявні пропуски. Тому функція **dlmread**, приведена вище, працюватиме правильно, якщо навіть вміст файлу ph.dat буде таким:

```
7.2; 8.5; 6.2; 6.6  
5.4; 9.2 ;8.1; 7.2
```

### Попередження.

Перший аргумент M-функції **dlmread** - це ім'я файлу, а не ідентифікатор файлу. Тому не потрібно заздалегідь відкривати файл за допомогою функції  **fopen**, а слід відразу застосовувати функції **dlmread** і **dlmwrite**.

Продемонструємо, як функція **dlmwrite** виконує запис тексту з роздільниками в зовнішній файл з ім'ям myfile, використовуючи роздільник ";":

```
A =  
1 2 3  
4 5 6  
dlmwrite('myfile',A,')  
1; 2; 3  
4; 5; 6
```

### Обмін файлами даних для різних платформ.

Іноді виявляється необхідно працювати з версіями системи MATLAB для різних обчислювальних платформ або передавати розроблені застосування на інші системи. Додатки, що створюються в системі MATLAB можуть включати M-файли, що є M-функціями або M-сценаріями, а також MAT-файли, що містять

двійкові дані. Обидва типи файлів можуть бути безпосередньо використані на різних платформах:

Mat-файли є двійковими файлами і залежать від типу використовуваного комп'ютера. Проте, вони можуть переноситися з одного типу комп'ютера на інший, оскільки містять ознаку використовуваного комп'ютера в заголовку файлу. Система MATLAB перевіряє цю ознаку, коли завантажує файл і, якщо виявляється, що файл створений на комп'ютері іншої платформи, виконує необхідне перетворення.

Щоб використовувати MATLAB на комп'ютерах різних платформ, необхідні програми обміну даними для двійкового і ASCII -формату. При використанні цих програм треба бути упевненими, що MAT -файли передаються як двійкові файли, M-файли - як ASCII -файли. Помилка в установці відповідних режимів зазвичай руйнує дані.

### **Команда diary.**

Ця команда дозволяє сформувати щоденник сеансу роботи, включаючи графічний вивід. Щоденник записується в спеціальний файл на жорсткому диску. Після сеансу роботи цей файл можна проглянути за допомогою будь-якого текстового редактора.

Наприклад, щоб створити в поточному каталозі файл щоденника з ім'ям febr01.out слід використовувати команду `diary febr01.out`. Для того, щоб в процесі ведення щоденника перервати запис, досить скористатися командою `diary off`, а для відновлення командою `diary on`.

### **M-код-файл Startup.**

Файл `matlabrc.m`, який розміщений в каталозі `local`, зарезервований для використання програмістами фірми MathWorks, а на розрахованих на багато користувачьких системах для використання менеджером системи. Файл `startup.m` призначений для користувача. У нім можна установити, шляхи доступу, що задаються за умовчанням, дескриптори графіки, а також змінні робочої області.

Наприклад, у файл `startup.m` можна ввести рядок, який додасть каталог `/home/me/mytools` до встановленого за умовчанням списку шляхів доступу `addpath /home/me/mytools`.

## **1.6. Використання пам'яті.**

Система MATLAB вимагає для зберігання кожної матриці безперервної області пам'яті. Зокрема, образи і анімація можуть споживати дуже великі об'єми пам'яті. На додаток до пам'яті для зберігання матриці, карта пікселів, використовувана для образів, вимагає пам'яті, пропорційної площі зображення. Так наприклад,

зображення 500?500 кольорових пікселів треба 2 МБ оперативної пам'яті. Якщо потрібно 10 зображень такого розміру, то вже необхідно 20 МБ, що є дуже великим об'ємом. Щоб зменшити об'єм пам'яті, потрібний для цих операцій, треба обмежити розмір зображень, що виводяться.

**Дозвіл проблем виділення пам'яті.** Якщо відсутній фрагмент пам'яті, достатній для розміщення матриці, то виникає помилка out of memory, хоча загальний об'єм вільної пам'яті може бути великим. Це пов'язано з фрагментацією пам'яті в процесі її виділення. Щоб ліквідувати фрагментацію, слід скористатися командою `pack`; інший спосіб - розмістити масиви великих розмірів в оперативній пам'яті заздалегідь на початку сеансу роботи.

## Управління динамічною пам'яттю.

Система MATLAB використовує для виділення динамічній пам'яті стандартні функції **malloc** і **free** мови C. Ці утиліти підтримують пул пам'яті, яка розподіляється операційною системою у відносно повільному темпі; у свою чергу, для системи MATLAB ця пам'ять виділяється набагато швидше. Якщо пул недостатній, то утиліт **malloc** запрошує операційну систему відносно виділення іншого фрагмента оперативній пам'яті, щоб поповнити пул. У міру виділення пам'яті пул може ставати дуже великим. Щоб підтримати швидкодію, утиліти **malloc** і **free** не повертають використану пам'ять операційній системі. Ці підпрограми виходять з припущення, що якщо великий об'єм пам'яті було потрібно один раз, то в ній виникне необхідність знову. Побічний ефект цього алгоритму полягає в тому, що якщо MATLAB використовував деякий об'єм пам'яті один раз, то вона не доступніша іншим програмам, навіть якщо MATLAB не використовує це. Пам'ять пулу повертається операційній системі лише після закінчення роботи системи MATLAB.

### 1.7. Інтерактивний доступ до довідкової інформації і документації. Команди **Help**, **lookfor**. Меню **Help**.

Існують наступні способи отримати інформацію про функції системи MATLAB в процесі сеансу роботи:

- команда **help**;
- команда **lookfor**;
- меню **Help**;
- перегляд і вивід на друк сторінок документації;
- звернення до web-сервера фірми The Mathworks.

#### Команда **Help**

Основний і найбільш швидкий спосіб з'ясувати синтаксис і особливості вживання M-функції - це використовувати команду **help** <ім'я M-функції>. Відповідна інформація з'являється безпосередньо в командному вікні. Наприклад, команда **help magic** виведе в командне вікно наступну інформацію англійською мовою.

#### **help magic**

MAGIC Magic square.

MAGIC(N) is an N-by-N matrix constructed from the integers 1 through N<sup>2</sup> with equal row, column, and diagonal sums. Produces valid magic squares for N = 1,3,4,5...

#### **MAGIC** Магічний квадрат.

MAGIC(N) - це матриця розміру  $N \times N$ , побудована з цілих чисел від 1 до  $N^2$  так, що суми елементів по рядках, стовпцям і діагоналям збігаються. Формує правильні магічні квадрати для  $N = 1, 3, 4, 5, \dots$ . Слід звернути увагу, що текст інтерактивної довідки використовує верхній регістр написання імен функцій і змінних, щоб виділити їх з останньої частини тексту. Проте при введенні імен функцій в командному рядку завжди використовуються символи нижнього регістра, а оскільки система MATLAB чутлива до вибору регістра, а дійсні імена функцій записуються рядковими буквами.

Всі функції системи MATLAB, а їх більше 800, організовані в логічні групи, і структура каталогів заснована на цій організації. Наприклад, всі функції лінійної алгебри знаходяться в каталозі `matfun`. Можна роздрукувати всі функції цього каталога з короткими поясненнями, якщо використовувати команду `help matfun`: **help matfun**

Matrix functions - numerical linear algebra.

Matrix analysis.

<code>norm</code>	- Matrix or vector norm.
<code>normest</code>	- Estimate the matrix 2-norm.

Linear equations.

<code>\ and /</code>	- Linear equation solution; use "help slash".
<code>inv</code>	- Matrix inverse.

Eigenvalues and singular values.

<code>eig</code>	- Eigenvalues and eigenvectors.
<code>svd</code>	- Singular value decomposition.

Matrix functions.

<code>expm</code>	- Matrix exponential.
<code>logm</code>	- Matrix logarithm.

Factorization utilities

<code>qrdelete</code>	- Delete column from QR factorization.
<code>qrinsert</code>	- Insert column in QR factorization.

Команда `help` сама по собі виводить на екран список каталогів

**help**

HELP topics:

`matlab\general` - General purpose commands.

`matlab\ops` - Operators and special characters.

`matlab\lang` - Programming language constructs.

`matlab\elmat` - Elementary matrices and matrix manipulation.



## Команда lookfor

Ця команда дозволяє виконати пошук М-функції по ключовому слову; при цьому аналізується перший рядок коментаря, і вона ж виводиться на екран, якщо в ній трапилося ключове слово. Наприклад, в системі MATLAB немає М-функції з ім'ям `inverse` і тому на команду `help inverse` відповіддю буде - `inverse.m not found`.

Проте команда `lookfor inverse` знайде не менше дюжини збігів, і це залежатиме від того, які ППП підключені до системи MATLAB.

### lookfor inverse

INVHILB Inverse Hilbert matrix.

ACOS Inverse cosine.

ACOSH Inverse hyperbolic cosine.

IFFTN N - dimensional inverse discrete Fourier transform.

IPERMUTE Inverse permute array dimensions.

ICCEPS Inverse complex cepstrum.

IDCT Inverse discrete cosine transform.

`idctold.m`: %IDCT Inverse discrete cosine transform.

Додавання опції - `all` команді `lookfor` у формі `lookfor шаблон - all` дозволяє переглядати усі рядки коментаря до М-функції, а не тільки перший рядок.

### Меню Help

Це меню командного вікна системи MATLAB показане на рис. 1.9 і дозволяє активізувати наступні вікна:

<b>Help Window</b>	Вікно довідки
<b>Help Tips</b>	Вікно довідки для отримання підказки
<b>Help Desk (HTML)</b>	Доступ до довідкових систем на жорсткому диску або CD - ROM
<b>Examples and Demos</b>	Вікно демонстраційної підсистеми
<b>About MATLAB</b>	Інформація про встановлену версію
<b>Subscribe (HTML)</b>	Підписка на послуги фірми The MathWorks, Inc.

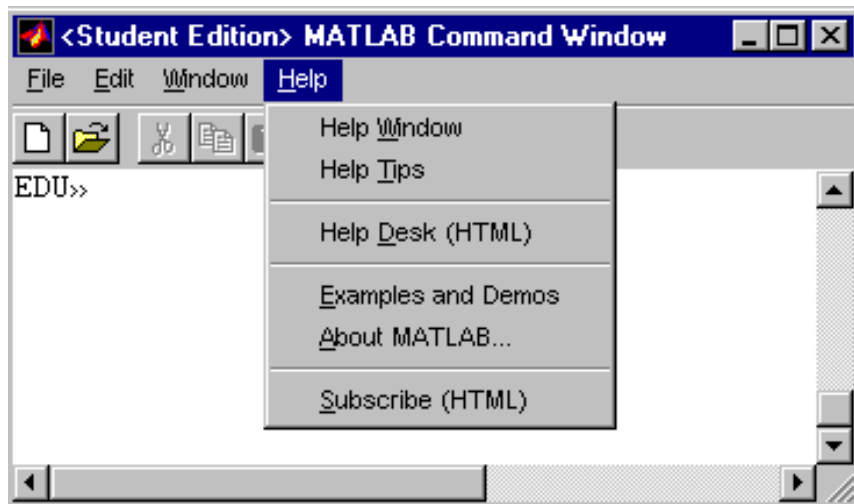


Рис. 1.9

### Вікно довідки Help Window.

Це вікно може бути викликане декількома способами: як опція вікно допомоги меню Допомога, натисканням кнопки? інструментальної панелі, або за допомогою команди `helpwin` (рис. 1.10).

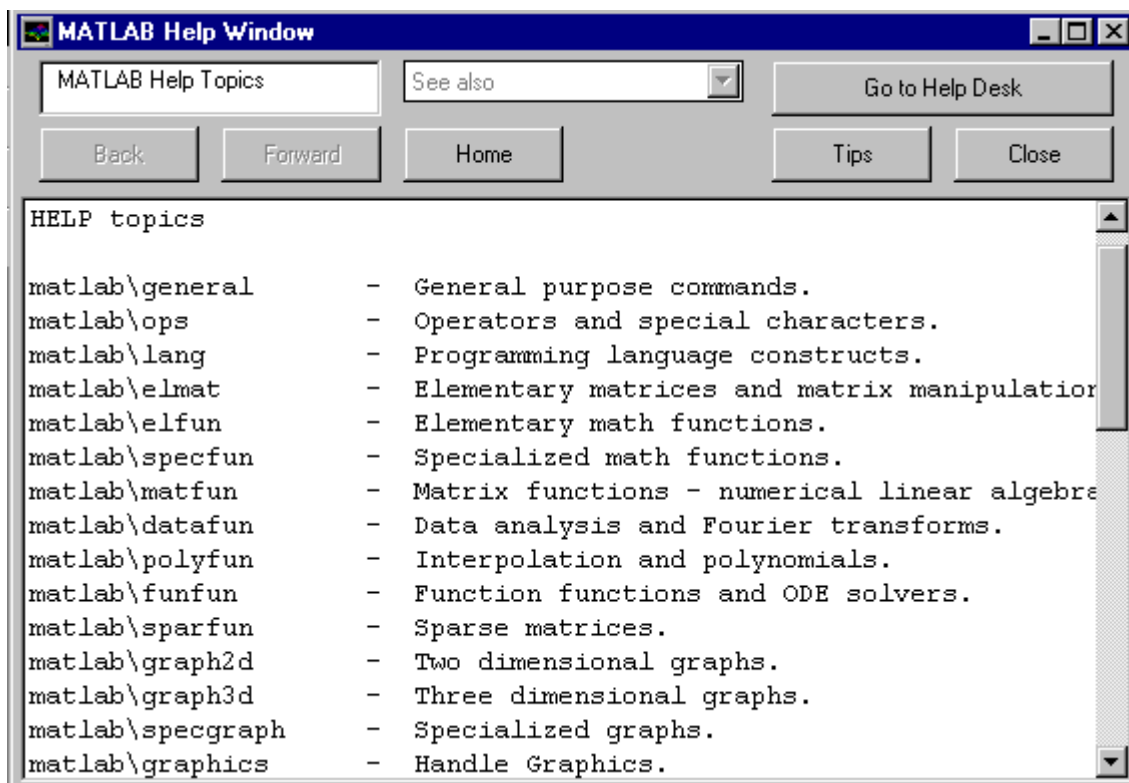


Рис. 1.10

На цьому малюнку показано початкове вікно MATLAB довідки; виділяючи будь-яку з рядків списку і двічі клацаючи лівою кнопкою миші, можна переходити до списків відповідних розділів. Така дія аналогічно команді `helpwin <ім'я розділу>`. Спадаючий список Див також в цьому випадку не активний. Кнопки Назад, Вперед,

Головна дозволяють переходити від одного активізованої вікна до іншого, або повернутися до початкового вікна.

У правому верхньому куті вікна розташовані 3 кнопки ( Go to Help Desk, Tips, Close).Перейдіть в розділ реєстрації, Поради, Закрити.

Кнопка (Tips) Поради виводить вікно підказок (рис. 1.11). У цьому вікні спадаюче меню активізовано і дозволяє виконати ряд додаткових довідкових команд (More help info help (HTML), lokfor, which, demo, general).

Більш допомоги Довідка (HTML), lokfor, які, демо, взагалі.

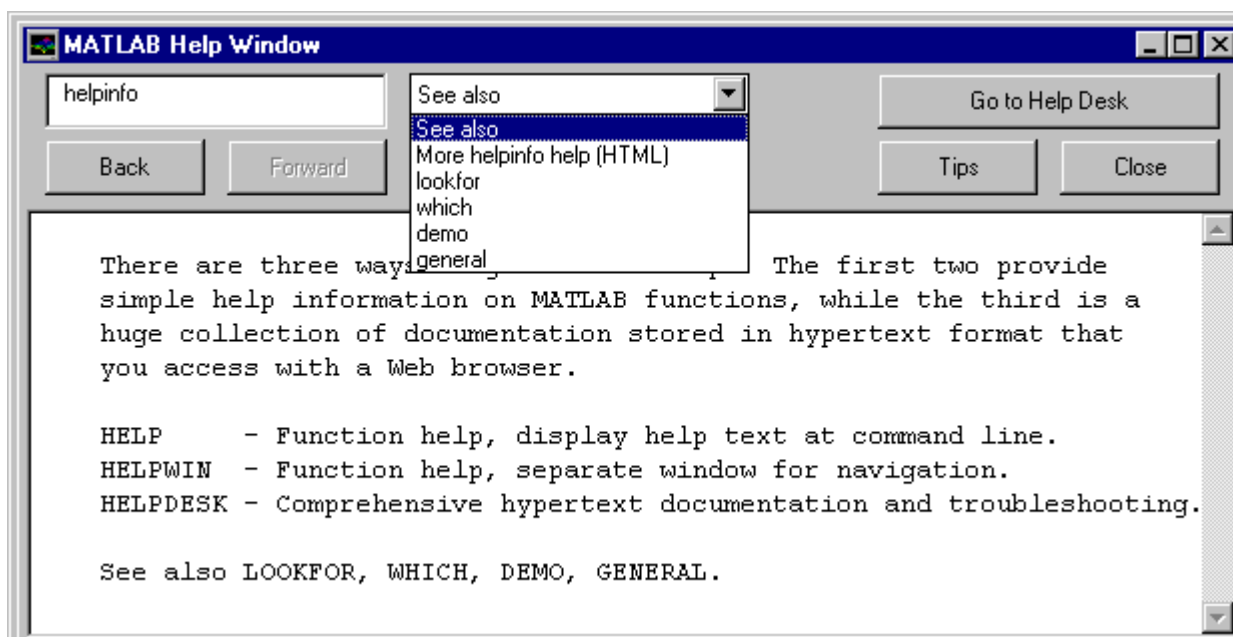


Рис. 1.11.

### Опція Help Desk .

Ця опція дозволяє отримати доступ до великого обсягу довідкової інформації та до документації по системі , що розміщується або на жорстких дисках , або на диску CD- ROM в рамках використовуваного персонального комп'ютера.

Багато з документів використовують мову гіпертекстових посилань HTML (мова розмітки гіпертексту ) і доступні для перегляду за допомогою таких засобів як Netscape Navigator або Microsoft Explorer .

Ця опція може бути також ініційована за допомогою команди підтримки.

Всі оператори і функції системи MATLAB описані у форматі HTML і містять більше подробиць і прикладів , ніж довідки по команді допомогу. Доступні HTML - версії різних документів , включаючи описи , керівництва користувача по системі і пакетам прикладних програм. Реалізована пошукова система дозволяє виконати необхідні команди.

**Опція Про MATLAB.** Ця опція виводить на екран заставку системи з вказівкою версії та приладдя користувачеві (рис. 1.12)

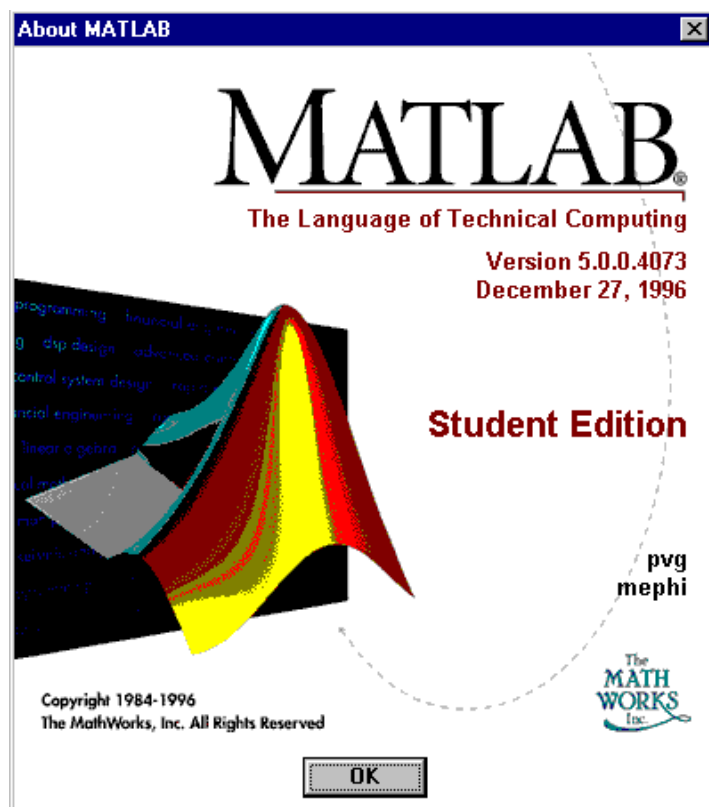


рис. 1.12

### **Перегляд і роздрук документації.**

Версії довідкової документації доступні для перегляду і роздрукування у форматі PDF (Portable Document Format) за допомогою кошти компанії Adobe Acrobat. Воно дозволяє переглядати текст у форматі друкованої сторінки, з повним набором шрифтів, графіки і зображень, з повним відчуттям читання книги. Одночасно це і найкращий спосіб отримання копій потрібних сторінок.

### **Веб -сервер фірми MathWorks .**

Якщо Ваш комп'юер приєднаний до мережі Інтернет , то можна реалізувати через механізм Help Desk з'єднання з WWW- сервером фірми- виробника і вийти на сторінку системи MATLAB. Можна також воспользоваться електронною поштою, щоб задати питання , внести пропозиції або повідомити про обнаружно помилках. Можна воспользоваться пошуковою системою WWW- сервера , щоб зробити запрос до постійно оновлюваної бази даних технічної підтримки. Можливості WWW- сервера постійно і розширюються і ви можете більш детально

ознайомитися з його інформаційними можливостями , безпосередньо зв'язавшись з його дзеркалом у Європі за адресою WWW- [europe.mathworks.com](http://europe.mathworks.com) .