التحليل العددى

مع التطبيق على الماتلاب

أ.د. محمد ابراهيم العدوى

قسم الإلكترونيات والاتصالات والحاسبات – كلية الهندسة بحلوان – جامعة حلوان

٢٠١٨

الإهداء

إلى كل من يحترم لغته ويعتز بها !!!

رجاء من قراء هذا الكتاب

الكتاب متاح لجميع القراء دون أى تكلفة وللاستفادة منه على أى وجه. فرجاء إذا رأيت عزيزى القارىء أنك قد استفدت منه فلا أطلب منك سوى الدعاء لمؤلفه إن كنت غير قادر ماديا، أما القارىء القادر ماديا فأطلب منه التبرع بما يستطيع لأى جهة خيرية يريد، وليكن على سبيل المثال مستشفى سرطان الأطفال ٥٧٣٥٧ بالقاهرة، أو مستشفى الكبد بالمنصورة، أو مستشفى القلب (مجدى يعقوب) بأسوان، أو هيئة مصر الخير، أو صندوق تحيا مصر، مع نية ثواب التبرع للمتبرع وللمؤلف.

المؤلف

أ.د.محمد ابراهيم العدوى

استعراض الكتاب

التحليل العددى

علم التحليل العددى يختلف على تعريفه الكثير، ولكن الأغلبية يتفقون على أنه علم استخدام الخوارزميات العددية للحصول على حلول تقريبية للمسائل الحسابية المعقدة، على العكس من الحلول التحليلية التى تعطى حلولا صحيحة تامة وغير تقريبية. لذلك فإن الهدف العام من علم التحليل العددى هو تصميم خوارزمات (طرق) تقريبية لحل المسائل الصعبة التى يصعب حلها بالورقة والقلم مثل أن تجد حلا لألف من المعادلات فى ألف من المجاهيل. أو تحسب معكوس مصفوفة مربعة من ألف صف وألف عمود مثلا.

على الرغم من امتداد أصول التحليل العددى فى التاريخ إلا أن الطفرة التى حدثت فى الحاسبات فى نهاية القرن الماضى كان لها أكبر الأثر فى استخدام هذا العلم فى الكثير من التطبيقات الهندسية والعلمية والطبية والحيوية وشتى نواحى العلوم التى نعرفها هذه الأيام. هذه الطفرة التى نعيشها الآن لن تمنعنا من إلقاء نظرة سريعة على تاريخ هذا العلم وبالذات على تاريخ كلمة خواريزم التى لا تخلو صفحة تقريبا من صفحات أى كتاب فى التحليل العددى من ذكرها. نبدأ أولا بالتعريف الرسمى لكلمة خواريزم من دائرة معارف بريتانيكا (الطبعة ١٥) التى تصف الخواريزم بأنه:

" Systematic mathematical procedure that produces - in a finite number of steps – the answer to a question or the solution of a problem."

أو بمعنى آخر فإن الخواريزم هو "خطوات حسابية منظمة تعطى ـ فى عدد محدد من الخطوات ـ إجابة لسؤال أو حلا لمشكلة".

وليس أفضل هنا من أن نضع بالنص مقدمة كتاب "تاريخ الخوارزمات من الحجر إلى الشريحة History of algorithms from the pebble to microchip" للمؤلف جين ليس تشيبيرت ورفاقه Jean Luc Chabert et al من الناشر سبرنجر ١٩٩٩. لقد بدأ الكاتب مقدمة كتابه بهذا النص العربى المصور:

> فأما الأموال والجذور التى تعدل العدد فمثل قولك
> مال وعشرة أجذاره يعدل تسعة وثلاثين درهما ومعناه أى مال اذا زدت عليه مثل عشرة أجذاره بلغ ذلك كله تسعة وثلاثين . فبابه(٢) أن تنصف الأجذار وهى فى هذه المسئلة خمسة فتضربها فى مثلها فتكون خمسة وعشرين فتزيدها على التسعة والثلاثين فتكون أربعة وستين فتأخذ جذرها وهو ثمانية فتنقص منه نصف الأجذار هو خمسة فيبقى ثلاثة وهوجذر المال الذى تريد والمال تسعة .

هل تعلم عما يتحدث هذا النص، إنه يصف خواريزم أو خطوات لحل معادلات الدرجة الثانية من خلال المثال $x^2+10x=39$. وهل تعلم من صاحب هذا النص، إنه محمد بن موسى الخوارزمى وهو من خواريزم بجنوب آسيا، وهو عالم فى الرياضيات فى القرن التاسع الميلادى وكان كتابه باسم "المختصر فى حساب الجبر والمقابلة" ولقد أطلق إسم الخواريزم algorithm نسبة لهذا العالم وهو المنشيء الأساسى لعلم الجبر algebra. إنه يصف حل هذه المعادلة فى الخطوات التالية:

١- إقسم معامل الجذر على اثنين وهذا يعطى ١٠÷٢=٥

٢- إضرب الخمسة فى نفسها ٥×٥=٢٥

٣- إجمع الخمسة وعشرون على الجانب الأيمن ٣٩+٢٥=٦٤

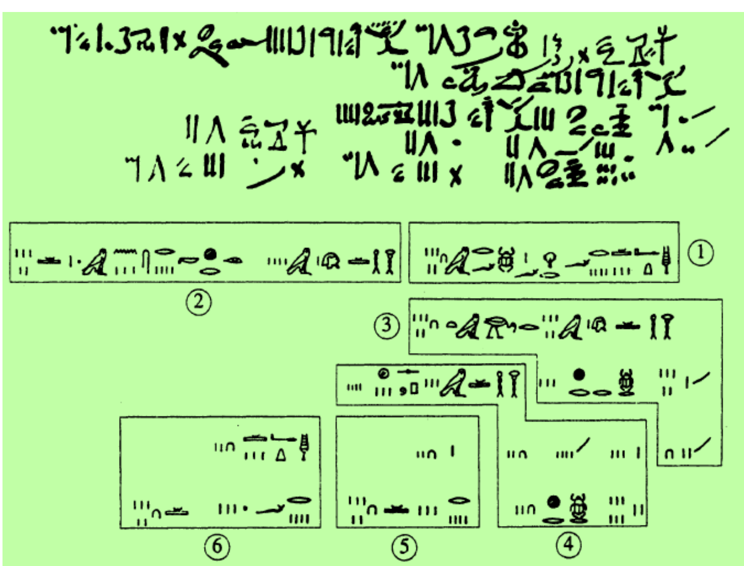
٤- أوجد الجذر التربيعى لهذا الناتج جذر ٦٤=٨

٥- إطرح من هذا الناتج نصف معامل الجذر الذى فى الخطوة ١، ٨-٥=٣

٦- هذا الناتج، ٣، هو حل هذه المعادلة.

خطوات مرتبة ومنظمة أعطت الحل فى النهاية. تذكر أننا هنا فى القرن التاسع الميلادى.

سنرجع للوراء قليلا ونغوص فى أعماق التاريخ لنجد فى بردية رهند Rhind papyrus التى كتبها الكاتب أحمس حوالى عام ١٦٥٠ قبل الميلاد وتحتوى البردية على حوالى مائة مسألة عبارة عن أمثلة حسابية تمت ممارستها فى ظروف خاصة. المسائل R24 حتى R27 فى هذه البردية تمثل أنواع حسابية مختلفة يمكن تمثيلها بالمعادلة التالية $x+(1/n)x=b$ حيث n تساوى ٧ و ٢ و ٤ و ٥. يقول مؤلف كتاب تاريخ الخوارزمات، "لقد اخترنا المسألة R26 من هذه البردية لأنها المسألة الأفضل شرحا. إنها تقابل حل المعادلة $x+(1/4)x=15$. فى النص المقابل نرى النص الأصلى الأغريقى وأسفله ترجمته المصرية القديمة. لقد استخدمنا البنط العريض للنص المكتوب أصلا بالحبر الأحمر. لقد استمر الكاتب فى شرح الخطوات أو الخوارزم algorithm الذى يحل المسألة السابقة والتى منطوقها حساب العدد الذى إذا أضيف إلى ربعه يعطى القيمة ١٥.



إذن فخواريزمات التحليل العددى ضاربة بجذورها فى أعماق التاريخ العربى والفرعونى وللأسف لا نجد كتابا دراسيا باللغة العربية يعتبر مرجعا شاملا لطلاب الجامعات والتطبيقين والمهتمين من التخصصات المختلفة. لذلك فهذا الكتاب يعتبر محاولة جادة فى هذا الطريق.

هذا الكتاب "التحليل العددى" موجه لطلاب السنوات قبل النهائية والنهائية من الجامعات ويمكن تدريسه على مدار فصل دراسى واحد فى صورة محاضرة من ساعتين دراسيتين أسبوعيا وأيضا تمارين (أو معمل حاسبات) لمدة ساعتين أسبوعيا يتم فيها حل بعض التمارين وكذلك تنفيذ بعض البرامج لتنفيذ الخواريزمات المقدمة فى الكتاب. لقد تم اختيار لغة الماتلاب MATLAB كلغة مصاحبة لهذا الكتاب والتى سيتم بها تنفيذ كل البرامج المقدمة نتيجة إمكانيات الماتلاب فى الرسم وعرض النتائج. هناك كتاب للمؤلف يحتوى شرحا مبسطا للماتلاب بالعربى على الموقع https://heshameladawy.com/my-dad-startup .

الخلفية العلمية المطلوبة لدارس هذا الكتاب هى الجبر الخطى بما فيه المتجهات والتعامل مع المصفوفات وحل المعادلات الخطية بالإضافة إلى التفاضل والتكامل وحل المعادلات التفاضلية. وهذه يتم دراستها فى السنوات الأولى من المرحلة الجامعية. بالإضافة إلى ذلك فإن الدارس لابد أن تكون لديه خبرة فى البرمجة بأى لغة من اللغات حتى ولو لم تكن لغة الماتلاب لأن الانتقال إلى البرمجة بالماتلاب ستكون عملية سهلة جدا لأى طالب لديه الخبرة عن البرمجة بأى لغة أخرى. ونحن فى هذا الكتاب سنقدم مراجعة للضرورى من لغة الماتلاب فى كل جزء وبصورة مختصرة لأننا نؤكد أن هذا الكتاب ليس كتابا عن البرمجة بالماتلاب ولكنه كتاب فى التحليل العددى. فى هذا المقام ننصح القارىء بالاستعانة بكتاب "تعلم ماتلاب بنفسك" لنفس المؤلف وآخرين للناشر جامعة الملك سعود بالمملكة العربية السعودية.

فصول الكتاب مرتبة كالتالى:

يقدم **الفصل الأول** مراجعة ضرورية وسريعة على ماتلاب وأوامره الكثيرة الاستخدام التى سنستخدمها فى هذا الكتاب وهذا يمثل كما قلنا مراجعة سريعة وبسيطة لهذه الأوامر بدلا من اللجوء إلى مرجع متخصص وبالذات أننا أضفنا فى نهاية الكتاب ملحقا يضم كل أوامر ماتلاب المستخدمة فى الكتاب مرتبة ترتيبا أبجديا مع شرح مبسط لكل منها، ونؤكد هنا أننا قدمنا الأوامر المستخدمة فى الكتاب فقط لأن أوامر ماتلاب كثيرة ولا حصر لها. بالطبع نحن نتوقع أن كثيرا من القراء سيكون لديهم فكرة جيدة عن ماتلاب، وفى هذه الحالة يمكنهم تخطى هذا الفصل والانتقال إلى الفصول الأخرى.

الفصل الثانى عبارة عن مراجعة على المصفوفات وكيفية إجراء العمليات المختلفة عليها فى ماتلاب وبالذات حساب معكوسها ومحددتها واستخدام طريقة كرامر لحل مجموعة من المعادلات الخطية وهو من المواضيع المهمة التى سيتم تغطيتها بتفصيل أكثر فى فصول أخرى. يقدم الفصل أيضا طريقة تحليل المصفوفة إلى مصفوفتين مثلثيتين علوية وسفلية واستخدام ذلك أيضا فى حل مجموعة من المعادلات الخطية.

الخطأ فى إجراء الحسابات عن طريق الحاسب الآلى يعتبر صغيرا جدا ويتم إهماله فى معظم التطبيقات ولكن عند إجراء الخوارزمات التى تنفذ فى صورة العديد من الحلقات فإن هذا الخطأ يتراكم ويصبح كبيرا يؤدى إلى كوارث. يقدم **الفصل الثالث** التعريف بالخطأ والدقة والانضباط والفرق بين كل منها والأنواع المختلفة للأخطاء وكيفية التقليل من آثارها.

يتناول **الفصل الرابع** الطرق المختلفة لحل المعادلات غير الخطية ذات المتغير الواحد التى على الصورة $f(x)=0$. هذه الحلول تسمى فى معظم الأحوال الجذور أو الأصفار لهذه المعادلة حيث بالتعويض بأحد هذه الجذور أو الأصفار فى التعبير $f(x)$ فإن النتيجة تكون صفرا. المعادلة $f(x)=0$ يكون لها فى العادة أكثر من جذر أو حل، والطرق التى سنقدمها فى هذا الفصل تبحث عن حل واحد معين فى نطاق معين للمتغير x.

توجد نظم المعادلات الخطية فى الكثير من التطبيقات الهندسية مثل الأبنية والتركيبات، والجوامد المرنة، والتدفق الحرارى، والمجالات الكهرومغناطيسية والدوائر الكهربية وغيرها الكثير، ولذلك سنقدم فى **الفصل الخامس** الطرق المختلفة لحل عدد n من المعادلات فى عدد n من المجاهيل على الصورة $Ax=b$ حيث A هى مصفوفة معاملات النظام و b هى مصفوفة الثوابت، و x هى مصفوفة المجاهيل المراد الحصول عليها.

على العكس من الفصل الخامس، فإن الطرق التكرارية التى يتم تقديمها فى **الفصل السادس**، والتى تسمى بالطرق غير المباشرة، تبدأ بتخمين حل معين، وبطريقة تكرارية تحاول هذه الطرق التحسين من هذا الحل الذى تم تخمينه فى خطوات متتالية حتى يصبح التغير فى الحل مع التقدم فى هذه الخطوات صغيرا، حيث عندها نوقف الخطوات ويكون هذا الحل الأخير هو الحل المقدم لمجموعة المعادلات، وهذه الطريقة لها بعض المميزات على الطريقة المباشرة يتم شرحها فى الفصل.

يقدم **الفصل السابع** لموضوع غاية فى الأهمية للكثير من المهندسن والعلوم التطبيقة على وجه العموم وهو موضوع الاستيفاء interpolation. الاستيفاء هو طريقة لإنشاء نقاط جديدة فى مدى متقطع من نقاط البيانات المعروفة. وببساطة فإن الاستيفاء هو رسم منحنى مستمر يمر بجميع نقط البيانات والتعبير عن هذا المنحنى بأى صورة تحليلية، وهذا هو ما سنراه بالتفصيل فى هذا الفصل.

ما أكثر ما يكون لدينا دوال يكون من الصعب تفاضلها أو تكاملها بالطرق التحليلية، أو أن هذه الدوال تكون معطاة فى صورة نقاط رقمية أو جداول أو مصفوفات، فى هذه الحالة يكون التفاضل العددى أو التكامل العددى هو البديل. يقدم **الفصل الثامن** العديد من دوال التفاضل والتكامل العددى مع شرح لهذه الطرق المختلفة والفروق بينها.

فى العديد من مشاكل الحياة العملية، تتم نمذجة هذه المشاكل فى صورة معادلة تفاضلية أو نظام من المعادلات التفاضلية، وعادة ما تكون هذه المعادلة التفاضلية التى تمثل نموذج المشكلة معقدة جدا بحيث يصعب إيجاد حل تحليلى صحيح لها، وفى هذه الحالة يتم اتباع واحدة من طريقتين لإيجاد حل تقريبى لهذه المشكلة: الطريقة الأولى هى محاولة تبسيط المعادلة التفاضلية إلى

صورة يمكن حلها تحليليا واستخدام هذا الحل الصحيح للمعادلة المبسطة ليمثل حل المشكلة الأصلية. والطريقة الثانية وهى التى سنتبعها فى هذا الفصل تحتوى على طرق عددية يمكن بها إيجاد تقريب عددى مباشر لحل المعادلة الأصلية، وهذه الطريقة بالطبع تعطى دقة أفضل فى التقريب من الطريقة الأولى. يقدم **الفصل التاسع** العديد من هذه الطرق مع شرح لكل منها على حدة وشرح لدوال ماتلاب التى تستخدم فى هذا المجال.

ينتهى الكتاب **بملحق** يحتوى كل دوال ماتلاب المستخدمة فى هذا الكتاب مع شرح مبسط لكل منها ومثال أو اثنين يوضحان هذه الدالة بحيث يمكن استخدام هذا الملحق كمرجع سريع لهذه الدوال. نؤكد أن هذا الملحق يحتوى دوال ماتلاب المستخدمة فى الكتاب وليس كل دوال ماتلاب لأن دوال ماتلاب لا يمكن استيعابها فى مثل هذا الكتاب.

عند تدريس هذا الكتاب كمقرر دراسى يمكن تتبع نفس ترتيب الفصول من الأول حتى التاسع كما يمكن إعادة ترتيب الفصول من الرابع حتى التاسع لأن كل فصل منها عبارة عن موضوع مستقل. فى حالة استخدام الكتاب كمرجع للموضوعات التى يحتويها فإنه يمكن اللجوء مباشرة إلى الفصل الخاص بالموضوع المطلوب حيث غالبا لن تكون هناك حاجة لمراجعات أخرى حيث أن كل فصل يمثل موضوعا قائما بذاته.

مع أجمل التمنيات بالتوفيق.

المؤلف

أ.د. محمد ابراهيم العدوى
أستاذ متفرغ بكلية الهندسة بحلوان – جامعة حلوان – حلوان – القاهرة
98eladawy@gmail.com
Mohamed_salama01@h-eng.helwan.edu.eg

المحتويات

الفصل ١

مقدمة عن برنامج ماتلاب

الفصل ١

مقدمة عن برنامج ماتلاب MATLAB

١-١ ما هو الماتلاب ؟

برنامج ماتلاب هو برنامج عالى المستوى والأداء للغة عالية المستوى والأداء أيضا وبالذات فى الحسابات والتطبيقات التقنية مثل الهندسة بجميع فروعها والرياضيات والفيزياء وغيرها. هذا البرنامج عبارة عن وسط برمجة سهل الاستخدام يمكنك من استخدام هذه اللغة فى أبسط صورها كآلة حاسبة يكون فيها التفاعل بين المستخدم والبرنامج من أسهل وأسرع ما يكون إلى وسط برمجة آخر يمكن للمستخدم فيه أن يكتب أى خواريزم أو برنامج يقوم بحل مشكلته ثم تنفيذ هذا الخواريزم واختباره بالكيفية التى يريدها مثله فى ذلك مثل لغات البرمجة الشهيرة مثل لغة C والجافا وغيرها، إلى إمكانية وضع البرنامج أو الخواريزم الذى صممته فى صورة دالة من دوال ماتلاب يمكنك إضافتها إلى مكتبة ماتلاب وتنفيذها بمجرد كتابة اسمها كأحد الدوال الداخلية فى الماتلاب. علاوة على كل ذلك فتوجد فى ماتلاب مكتبة من دوال الرسم والمحاكاة التى تمكنك من عرض نتائجك كصور ثنائة الأبعاد وثلاثية الأبعاد بسهولة تعجز عنها الكثير من البرمجيات فى هذا المجال. إن استخدام ماتلاب كآلة حاسبة أو فى صورته التفاعلية تعتبر من أهم مميزات ماتلاب والتى لا توجد فى أى لغة برمجة أخرى حيث يمكن مباشرة حساب الجذر التربيعى مثلا لأى ثابت كما يمكن أيضا فى هذه الصورة التفاعلية إجراء الكثير من العمليات الحسابية المعقدة مثل حل المعادلات التفاضلية وبالطبع ستحتاج لأكثر من سطر.

بجانب كل ذلك يحتوى ماتلاب على الكثير من المكتبات النوعية أو المتخصصة يسميها الماتلاب صندوق الأدوات tool box تحتوى كل منها على الكثير من الدوال المتخصصة فى مجال معين والتى تسهل على المستخدم التعامل معها من خلال هذه المكتبات المتخصصة. من هذه المكتبات ما يلى:

- مكتبة المعلوماتية الطبية Bioinformatics tool box
- مكتبة نظم الاتصالات Communication systems tool box
- مكتبة نظم التحكم Control systems tool box
- مكتبة اكتساب أو قراءة البيانات Data acquisition tool box
- مكتبة قواعد البيانات Data base tool box
- مكتبة تصميم المرشحات Filter design tool box
- مكتبة المنطق الهلامى Fuzzy logic tool box

- مكتبة الخواريزمات الجينية Genetic algorithms tool box
- مكتبة معالجة الصور Image processing tool box

وهناك الكثير من هذه المكتبات التى يذخر بها هذا البرنامج والتى تعتبر من أهم مميزاته نظرا لما تقدمه من سهولة فى البرمجة والتعامل للمتخصصين فى كل هذه المجالات .

يحتوى برنامج ماتلاب على وسط برمجة آخر يستخدمه كل المهندسين وربما غيرهم أيضا وهو وسط برمجة المحاكاة Simulink . هذا الوسط يتعامل ليس من خلال أوامر وخواريزمات مكتوبة ولكن من خلال صناديق أو بلوكات محاكاة. فيمكنك مثلا سحب صندوق مصدر إشارة تتحكم فى تردد ومقدار هذه الإشارة من خلال خواصه، ثم تسحب صندوقا آخر يمثل مرشح تتحكم أيضا فى خواصة حيث يقوم هذا الصندوق بالسماح لبعض الترددات بالمرور للخرج ومنع البعض الآخر، ثم فى النهاية يمكنك أن تسحب صندوقا ثالثا يمثل سماعة تسمع من خلالها، على سماعة الحاسب الذى تتعامل معه، الصوت الخاص بالإشارة الخارجة من مصدر الإشارة حيث يتم ذلك بالطبع بعد توصيل هذه البلوكات مع بعضها على التوالى. كان هذا مجرد مثال بسيط باستخدام ثلاثة بلوكات يعرفها الجميع، ولكن بالطبع يمكن تصميم أنظمة كاملة بهذه الطريقة ودراسة أدائها فى هذا الوسط قبل البدأ فى تنفيذها. يحتوى ماتلاب على العديد من المكتبات المتخصصة فى كل المجالات فى هذه الصورة من بلوكات المحاكاة. بالطبع فإن وسط برمجة المحاكاة لا مجال لشرحه فى هذه المقدمة بالإضافة إلى أننا لن نستخدمه فى هذا الكتاب.

كلمة ماتلاب مأخوذة من التعبير "معمل المصفوفات Matrix Laboratory, MATLAB" وذلك لأن وحدة البيانات الأساسية فيه هى المصفوفة ، فكل المتغيرات وكل الثوابت ينظر إليها الماتلاب على أنها مصفوفة ، حتى أنك لو كتبت x=10 فإن ذلك يتم النظر إليه فى جميع لغات البرمجة على أن هناك متغير x وضعت فيه القيمة أو الثابت 10 ، ولكن فى الماتلاب يكون الوضع مختلف حيث ينظر الماتلاب للتعبير x=10 على أنه هناك متغير x وضعت فيه المصفوفة أحادية الأبعاد ، أى مصفوفة من صف واحد وعمود واحد ، أى مصفوفة بها عنصر واحد هو الثابت 10 . هذه الطريقة الفريدة لتمثيل البيانات التى اتبعها الماتلاب سمحت بحل الكثير من المشاكل بالذات التقنية منها والتى تتعامل مع مصفوفات بيانات بأبعاد هائلة يصعب معها التعامل أو تأخذ وقتا كبيرا لحسابها ومن أمثلة ذلك الصور . فتخيل مثلا أن لديك صورتين أبعاد كل منهما ١٠٢٤×١٠٢٤ بكسل وتريد ضرب هاتين الصورتين فى بعضهما ، إن ذلك يتم فى الماتلاب بسهولة جدا بالمقارنة باللغات الأخرى .

يتكون برنامج الماتلاب كنظام برامجى software system من خمسة أجزاء أساسية نذكرها فيما يلى :

١- **وسط البرمجة :** وهو مجموعة من الأدوات السهلة الاستخدام عن طريق النقر بالفأرة وهو أول شاشة تظهر لك على سطح المكتب بمجرد النقر على أيقونة الماتلاب والدخول فيه وهذه الشاشة

مقسمة إلى أكثر من جزء منها جزء خاص بمساحة التشغيل work space ، ونافذة الأوامر command window ، ومساحة تاريخ الأوامر command history وغيرها والتى سيأتى شرحها بالتفصيل فى الجزء التالى من هذا الفصل .

٢- **مكتبة دوال ماتلاب :** وهذه مكتبة كبيرة تضم العديد من الدوال سابقة التجهيز التى يتم تنفيذها بمجرد كتابة إسمها وتغطى هذه المكتبة تقريبا جميع التخصصات العلمية بحيث أنه ما من دالة تريد إجراءها فى معظم التخصصات العلمية إلا ستجدها جاهزة فى مكتبة ماتلاب وما عليك لكى تنفذها إلا أن تكتبها بصورتها الصحيحة . تتغير هذه الدوال من الدوال البسيطة مثل المجموع والمتوسط والجيب sine وجيب التمام cosine والتعامل مع الأرقام المركبة complex numbers إلى الدوال المعقدة مثل ضرب المصفوفات وإيجاد معكوسها ودوال بيسيل Bessel ومحولات فورير Fourier وغير ذلك الكثير.

٣- **لغة برمجة ماتلاب** : وهى لغة عالية المستوى وحدتها هى المصفوفة تشمل كل العناصر المطلوبة لأى لغة برمجة مثل الشروط والحلقات وهياكل البيانات والإدخال والإخراج والبرمجة الموجهة بهدف object oriented programming مثلها فى ذلك مثل لغة C والجافا حيث باستخدام هذه اللغة يمكن تصميم وبناء أعقد الخوارزمات.

٤- **مكتبة الرسم** : تضم هذه المكتبة العديد من الأوامر التى يمكن بها رسم أى دالة أو نتيجة فى الأبعاد الثنائية أو الثلاثية بجانب أوامر قراءة ومعالجة وعرض الصور. كل هذا بجانب تصميم شاشات التفاعل مع المستخدم Graphical user interface, GUI.

٥- **برنامج تفاعل ماتلاب مع التطبيقات** The matlab Application Program Interface, API: وهذه المكتبة تسمح للمستخدم بكتابة برامج بلغة C أو الفورتران وتوصيلها أو تنفيذها من خلال ماتلاب .

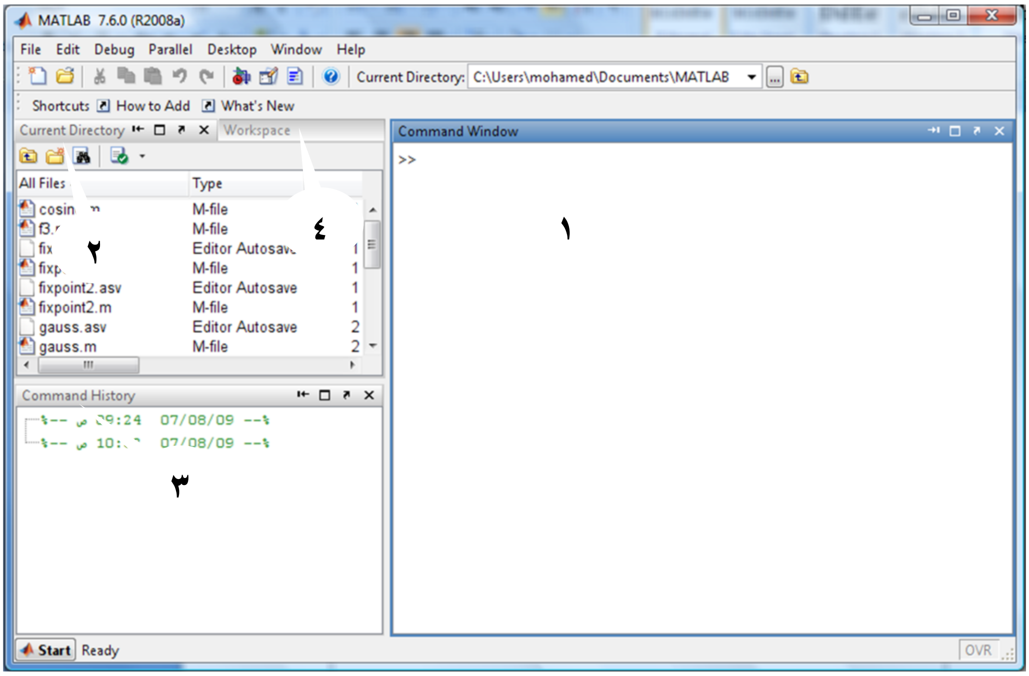
١-٢ بدأ التشغيل وسطح المكتب فى ماتلاب

يوجد الماتلاب فى العديد من الإصدارات حيث يظهر إصدارا جديدا كل عام تقريبا، ولا يوجد فرق كبير فى التعامل مع هذه الإصدارات وبالذات الأخيرة منها ولن نضيع وقتا فى شرح الفروق بينها.

من على سطح المكتب لبرنامج النوافذ أنقر على أيقونة ماتلاب مرتين ستدخل فورا فى شاشة وسط البرمجة الخاص به كما فى شكل (١-١) وهذه الشاشة سنطلق عليها سطح مكتب الماتلاب وهذه الشاشة موجودة فى كل الإصدارات تقريبا.

للخروج من ماتلاب بعد الانتهاء من جلسة برمجة يمكنك النقر على الخيار Exit من قائمة الملفات File من على سطح المكتب. يمكنك الخروج أيضا من ماتلاب بكتابة الأمر quit فى نافذة الأوامر على سطح المكتب .

تتكون شاشة سطح المكتب فى ماتلاب فى الغالب من ٤ أجزاء أو نوافذ رئيسية كما فى شكل (١-١) (ومن الممكن أن تكون هناك نوافذ أخرى على حسب وضع مكونات سطح المكتب) وهى كما يلى :



شكل (١-١) مكونات شاشة سطح مكتب الماتلاب

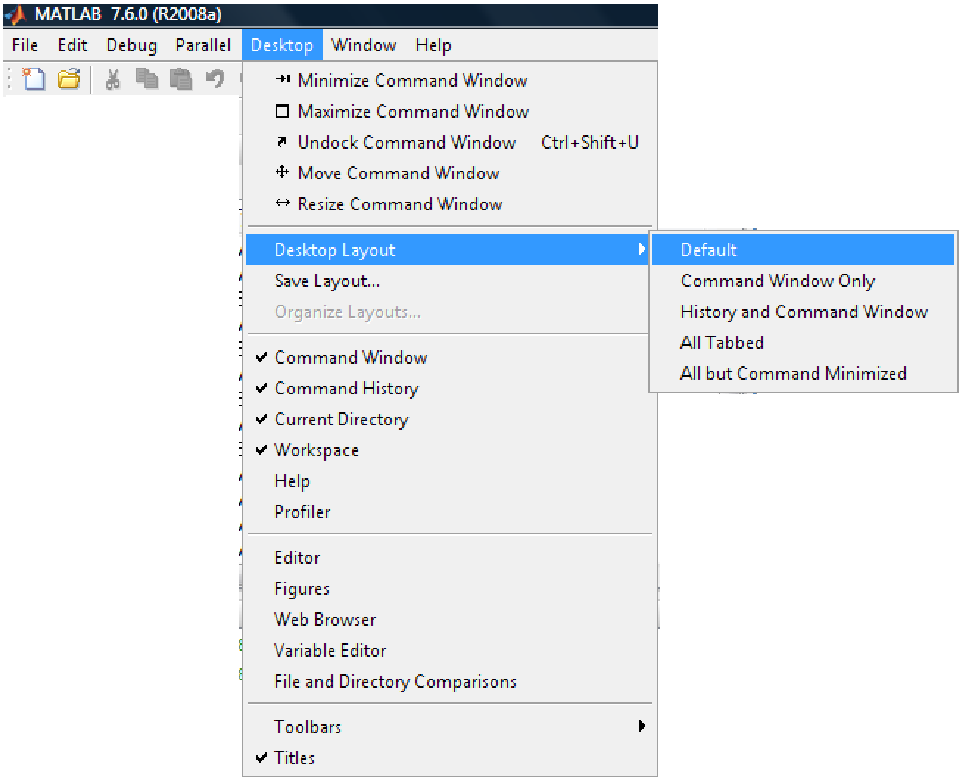
١- **نافذة الأوامر Command window** : وفيها نكتب الأوامر فى حالة استعمال ماتلاب فى الصورة التفاعلية أو كآلة حاسبة وتظهر فيها أيضا نتيجة تنفيذ أى أمر كما سنرى بعد قليل .

٢- **نافذة المجلد الحالى Current directory** : حيث تحتوى هذه النافذة محتويات المجلد الموجود فيه الماتلاب . يمكن تغيير هذا المجلد باستخدام السهم النازل فى شباك المجلد الحالى Current directory فى شريط الأيقونات . بالنقر مرتين على أى ملف فى هذا المجلد يتم فتحه .

٣- **نافذة تاريخ الأوامر Command history** : وتحتوى هذه النافذة على قائمة بكل الأوامر التى تم استخدامها فى نافذة الأوامر من بداية جلسة تشغيل ماتلاب حتى نهاية الجلسة . هذه االأوامر تكون مرتبة وبالنقر على أى أمر فيها مرتين يظهر هذا الأمر فى نافذة الأوامر ويتم تنفيذه. سنتحدث عن ذلك مع الحديث بشيء من التفصيل عن نافذة الأوامر فى الجزء التالى.

٤ – نافذة مساحة العمل Work space : هذه النافذة تظهر بالتبادل مع نافذة المجلد الحالى أحيانا حيث بالنقر على شريط مساحة العمل تظهر نافذة مساحة العمل وتختفى نافذة المجلد الحالى، وبالنقر على شريط المجلد الحالى مرة أخرى تظهر نافذة المجلد الحالى مرة أخرى وتختفى نافذة مساحة العمل. نافذة مساحة العمل تعرض جميع المتغيرات variables التى تم استخدامها فى أى جلسة تنفيذ من جلسات الماتلاب. إننا نعنى بكلمة جلسة هنا من بداية تشغيل ماتلاب حتى الخروج منه فى أى مرة من المرات.

نوافذ سطح المكتب للماتلاب الموجودة فى شكل (١–١) هى التقسيمة التلقائية default لهذه النوافذ. يمكن تصغير مساحة كل نافذة أو تكبيرها أو إخفاء بعضها باستخدام الفأرة كما نفعل مع أى نافذة فى برنامج النوافذ كما يمكن التحكم فى ترتيب هذه النوافذ وطريقة ظهورها بالكيفية التى يريدها المستخدم. بمجرد خروج المستخدم من أى جلسة وإغلاق ماتلاب فإن آخر وضع لهذه النوافذ يتم الاحتفاظ به.



شكل (١– ٢) اختيار تقسيمة سطح مكتب الماتلاب

يمكن التحكم فى كيفية ظهور سطح المكتب فى ماتلاب بالنقر على قائمة Disktop ومنها نختار Disktop Layout ثم نختار منها أى شكل نريد كما فى شكل (١– ٢). لاحظ أن الاختيار Default هو الاختيار

التلقائى الذى يختاره لك ماتلاب وهو الموضح فى شكل (١-١) وهو أفضل الاختيارات للكثير من المستخدمين. بالنقر على الاختيار Command Window Only ستظهر نافذة أوامر فقط ، والنقر على History and Command Window يظهر نافذة الأوامر ونافذة تاريخ الأوامر فقط ، وبالنقر على All Tabbed ستظهر لك أحد هذه النوافذ تملأ شاشة الحاسب مع أزرار tabs للنوافذ الأخرى بحيث يمكن التنقل بين هذه النوافذ بالنقر على الزرار الخاص بكل منها. حاول تجربة ذلك. فى أى لحظة يمكنك النقر على الاختيار التلقائى Default للعودة إلى الشكل التلقائى كما فى شكل (١-١). بالطبع ستكون هناك اختلافات بسيطة من إصدار لإصدار آخر ولكن مضمون ما نقول لابد أن يكون موجودا فى كل الإصدارات.

١-٣ الماتلاب التفاعلى أو الماتلاب كآلة حاسبة

سنرى فى هذا الجزء طريقة من طرق تشغيل ماتلاب والتى يتميز بها عن جميع لغات البرمجة وهى تشغيله كآلة حاسبة أو بطريقة تفاعلية حيث بمجرد كتابة الأمر والضرب على زرار Enter يتم تنفيذ الأمر وإظهار النتيجة مباشرة. لكى نرى ذلك، إبدأ فى تشغيل ماتلاب وانقر على نافذة الأوامر لكى تنفذ الأوامر التالية: (جميع الأوامر والعمليات التالية تم تنفيذها على ماتلاب ونسخها هنا ، ولذلك فإنك لو نفذتها بنفس الطريقة ونفس التتابع فلابد أن تحصل على نفس النتائج. الكتابة باللون الأزرق هى استجابة ماتلاب للأمر أو العملية المطلوبة بعد الدليل >>)

العمليات الحسابية البسيطة

```
>> 2+3
ans =
    5
```

العلامة >> معناها أن ماتلاب ينتظر أمر بعد هذه العلامة، وقد قمنا بكتابة 2+3 أى أن الأمر هو جمع هذين الرقمين، وبمجرد أن ندوس الزرار Enter سيرد عليك ماتلاب بالنتيجة ans=5 كما رأينا . فيما يلى سنكتب كل الأوامر بالخط العادى، والإجابة من ماتلاب سنكتبها بالخط العادى واللون الأزرق .

```
>> 3-2
ans =
    1
>> 3*2
ans =
    6
```

لاحظ فى الأمر السابق أن علامة الضرب هى *.

```
>> 3/2
ans =
   1.5000
```

وعلامة القسمة هى الشرطة المائلة ناحية اليمين /، هنا البسط على يسار العلامة والمقام على يمينها وبالتالى فإن الأمر السابق سيقسم الرقم 3 على الرقم 2 والنتيجة كانت 1.5.

```
>> 2\1
ans =
   0.5000
```

الشرطة المائلة ناحية اليسار تمثل عملية قسمة أيضا بحيث أن البسط هنا يكون على يمين العلامة والمقام على يسارها، وعلى ذلك فالأمر السابق يقسم 1 على 2 لتكون النتيجة 0.5.

```
>> 2^3
ans =
    8
```

العلامة ^ هى علامة الأس، أى أن الأمر السابق يحسب 2 أس 3 لتكون النتيجة 8.

- فى أثناء كتابة أى أمر يمكنك استخدام جميع أزرار تصحيح النص مثل أزرار السهم الأيمن right arrow والسهم الأيسر والزرار Del والزرار Backspace. لاحظ أن هذه الأزرار تساعد فى التصحيح على خط أو سطر واحد فقط فليس هناك تحرك لأعلى أو لأسفل مثلا باستخدام السهم لأعلى أو السهم لأسفل.
- السهم لأعلى والسهم لأسفل يمكن استخدامها لإحضار أى أمر سبق كتابته بدلا من أن نعيد كتابته مرة أخرى. فمثلا إذا كنت قد كتبت الأوامر السابقة بنفس الترتيب السابق فإنه بضرب زرار السهم لأعلى سيحضر لك الأمر 2^3، وبضرب الزرار Enter يتم تنفيذه. بضرب زرار السهم العلوى مرة أخرى يظهر أمامك الأمر 2\1، وإذا ضربت Enter يتم تنفيذه. وهكذا باستخدام السهم الأعلى والأسفل يمكن التجول فى كل الأوامر السابقة حيث بعد ظهور أى أمر تريده يمكنك تنفيذه بضرب الزرار Enter.
- ماتلاب به خاصية جميلة وهى الاستدعاء الذكى للأوامر السابقة، فبمجرد كتابة الأحرف الأولى من أى أمر ثم نضرب زرار السهم العلوى يظهر الأمر بالكامل حيث يمكنك تنفيذه بضرب الزرار Enter. مثلا فى الأوامر السابقة لو كتبنا +2<<، ثم ضربنا السهم العلوى فإن الأمر تكتمل كتابته فيصبح 2+3<< وبضرب Enter تظهر النتيجة.
- بمناسبة القسمة ماذا سيكون موقف ماتلاب عندما نقسم على الصفر. إنظر للأمر التالى :

```
>> 1/0
ans =
```

```
  Inf
```

عند قسمة واحد على صفر كانت نتيجة القسمة هى Inf وهى اختصار لمالانهاية Infinity كما نعلم.
هناك نتيجة أخرى قد تقابلها عند التعامل مع ماتلاب. أنظر للمثال التالى :

```
>> 0/0
ans =
  NaN
```

هنا نقسم صفر على صفر والنتيجة بالطبع غير محددة Not a Number, NaN كما رأينا.

المتغيرات

المتغير فى أى لغة برمجة هو عنوان فى الذاكرة يحمل اسم المتغير ويمكن أن نضع أى قيمة فى هذا المتغير باستخدام العلامة =. نبدأ الآن بإجراء هذه الأوامر على ماتلاب حيث قمنا بنسخها هنا حتى يستطيع القارىء إعادتها والتدريب عليها:

```
>> a=2
a =
   2
```

بمجرد كتابة a=2 ثم نضرب Enter يرد الماتلاب بالإجابة وهى أيضا a=2 مما يعنى أنه قد تم تخصيص القيمة 2 للمتغير a، ويمكن الاستفادة بها كما فى الأوامر التالية:

```
>> a+9
ans =
   11
```

هنا تم جمع الرقم ٩ مع المتغير a وكانت الإجابة 11،

```
>> a*9
ans =
   18
```

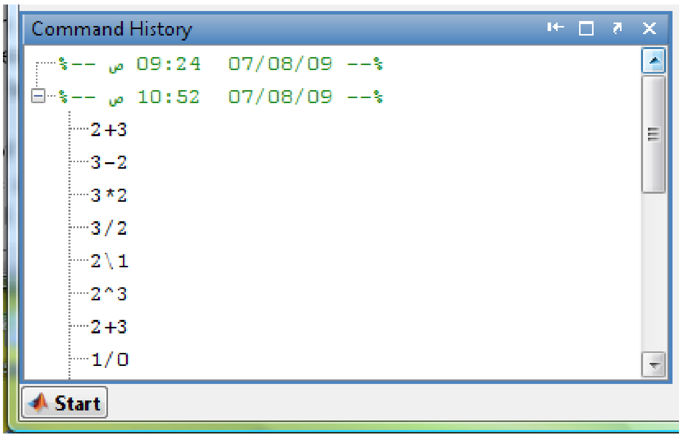
وهنا تم ضرب a فى 9 وكانت الإجابة 18. لاحظ أن قيمة المتغير a لازالت تساوى 2 ولم تتغير حتى الآن بالرغم من إجراء كل هذه العمليات السابقة على المتغير a. أنظر لهذا الأمر:

```
>> a=a+10
a =
   12
```

الآن تغيرت قيمة a وأصبحت a=12. أكتب الأمر التالى:

```
>> b=4;
```

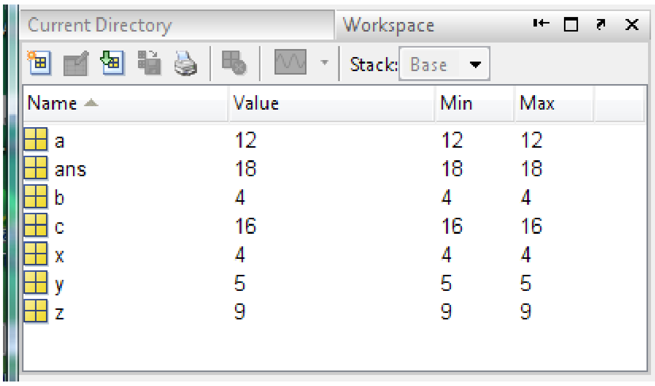
لاحظ أن الأمر b=4 هنا متبوعا بالفاصلة المنقوطة . وجود فاصلة منقوطة بعد أى أمر تمنع من عرض النتيجة فقط، بمعنى أنه يتم التنفيذ وتصبح قيمة المتغير b تساوى 4 ولكن لم تعرض النتيجة كما كان فى الأمر a=2.



شكل (١ – ٣) نافذة تاريخ الأوامر

```
>> c=a+b
c =
 16
```

آخر قيمة للمتغير a كانت 12 وبجمعها مع المتغير b تصبح النتيجة 16 التى يتم وضعها فى المتغير c وعرضها كما رأينا. أنظر للأمرين التاليين:



شكل (١ – ٤) نافذة مساحة العمل work space

لاحظ الأمرين التاليين:

```
>> b
b =
   4
>> B
??? Undefined function or variable 'B'.
```

الأمر الأول b<< عرض قيمة المتغير b الموجودة عنده وهى 4، بينما الأمر B<< لم يعرض أى نتيجة بل قال أن المتغير B متغير أو دالة غير محددة. إن هذا يقودنا إلى حقيقة مهمة وهى أن ماتلاب حساس لشكل الحرف. أى أن المتغير b يختلف عن المتغير B، كل منهما يعتبر متغير مختلف، وذلك على العكس من بعض لغات البرمجة التى لا تفرق بين الأحرف الصغيرة والكبيرة فصورة الحرف فى هذه اللغات ليس لها قيمة.

استخدام الفاصلة المنقوطة تمكنك من كتابة أكثر من أمر فى نفس السطر كما يلى:

```
>> x=4;  y=5;  z=x+y;
>> z
z =
    9
```

فى السطر الأول تم وضع 4 فى المتغير x، و 5 فى المتغير y، ومجموع المتغيرين x و y فى المتغير z، ولم يتم عرض أى نتيجة بسبب وجود الفاصلة المنقوطة فى نهاية كل أمر. بمجرد كتابة z تم عرض النتيجة z=9.

الآن يمكنك النظر إلى نافذة تاريخ تتابع الأوامر كما فى شكل (١– ٣) . تلاحظ فى هذه النافذة أن جلسة ماتلاب هذه كانت يوم ٢٠٠٩/٨/٧ الساعة العاشرة واثنين وخمسون دقيقة صباحا ثم بعد ذلك تم عرض جميع هذه الأوامر بنفس طريقة كتابتها فى نافذة الأوامر ولكن طبعا وكما نرى فى الشكل دون عرض نتيجة تنفيذ هذه الأوامر. بالنقر على أى أمر فى هذه النافذة يتم نقله إلى نافذة الأوامر وتنفيذه فحاول تجربة ذلك.

شكل (١– ٤) يبين نافذة مساحة العمل work space حيث تعرض هذه النافذة جميع المتغيرات التى تم استخدامها فى جلسة البرمجة الحالية مرتبة بحسب استخدامها فى نافذة الأوامر. هذه النافذة تعرض أيضا آخر قيمة أخذها كل متغير وأقل قيمة وأكبر قيمة أيضا لهذا المتغير.

نلاحظ من هذا الشكل أيضا أن نتيجة أى عملية حسابية التى تكتب على الصورة ans=5 مثلا، أن ans هو متغير أيضا من إنشاء ماتلاب ويمكن التعامل معه كأى متغير كما يلى:

```
>> x=4; y=6;
>> x*y
ans =
   24
>> ans=ans/2
ans =
   12
```

الدوال الحسابية

برنامج ماتلاب لديه مكتبة ذاخرة بالدوال الحسابية التى قد لا تخطر على بالك، ويمكن النداء عليها وتنفيذها ببساطة كما سنرى:

بالنسبة للدوال المثلثية مثل sin و cos فإن ماتلاب يتوقع وضع الزوايا بالراديان radian (التقدير الدائرى) وليس بالدرجات فمثلا الزاوية 30 درجة تكتب على الصورة 30*pi/180=pi/6 حيث pi=π.

```
>> sin(pi/6)
ans =
   0.5000
>> cos(pi/6)
```

```
ans =
   0.8660
```

اللوغاريتمات للأساس e :

```
>> log(10)
ans =
   2.3026
```

جدول ١–١ قائمة ببعض الدوال الحسابية الشهيرة

مسلسل	شرح الدالة		الدالة كما فى ماتلاب
١	جيب التمام العكسى ، النتيجة بالتقدير الدائرى	Inverse cosine; result in radians	acos
٢	جيب التمام العكسى ، النتيجة بالدرجات	Inverse cosine; result in degrees	acosd
٣	الجيب العكسى ، النتيجة بالتقدير الدائرى	Inverse sine; result in radians	asin
٤	الجيب العكسى ، النتيجة بالدرجات	Inverse sine; result in degrees	asind
٥	الظل العكسى ، النتيجة بالتقدير الدائرى	Inverse tangent; result in radians	atan
٦	الظل العكسى ، النتيجة بالدرجات	Inverse tangent; result in degrees	atand
٧	جيب التمام ، النتيجة بالتقدير الدائرى	Cosine of argument in radians	cos
٨	جيب التمام ، النتيجة بالدرجات	Cosine ofo argument in degrees	cosd
٩	الجيب ، النتيجة بالتقدير الدائرى	Sine of argument in radians	sin
١٠	الجيب ، النتيجة بالدرجات	Sine of argument in degrees	sind
١١	الظل ، النتيجة بالتقدير الدائرى	Tangent of argument in radians	tan
١٢	الظل ، النتيجة بالدرجات	Tangent of argument in degrees	tand
١٣	الأس الطبيعى e	Exponential	exp
١٤	اللوغاريتم الطبيعى للأساس e	Natural logarithm	log
١٥	اللوغاريتم للأساس 10	Common (base 10) logarithm	log10
١٦	الجذر التربيعى	Square root	sqrt
١٧	القيمة المطلقة	Absolute value	abs
١٨	التقريب لأعلى رقم صحيح	Round toward infinity	ceil
١٩	التقريب لأقل رقم صحيح	Round toward zero	fix
٢٠	التقريب لأقل رقم صحيح	Round toward minus infinity	floor
٢١	الباقى من القسمة	Remainder after division	rem
٢٢	التقريب لأقل رقم صحيح	Round to nearest integer	round

بينما اللوغاريتم للأساس 10 فيكتب على الصورة:

```
>> log10(10)
ans =
```

1

يحتوى ماتلاب على االعديد من الدوال الحسابية التى نقدم بعضها وأكثرها استخداما فى جدول ١-١ فحاول تجربة هذه الدوال أو على الأقل ما يخصك منها.

يمكن الحصول على قائمة كاملة بهذه الدوال من قائمة المساعدة Help ثم الدخول على MATLAB ثم Function Reference ثم Mathematics ثم Elementary Math حيث سترى عرضا كاملا للكثير من الدوال المستخدمة فى ماتلاب مبوبة وظيفيا وأبجديا.

طلب المساعدة فى ماتلاب

ماتلاب لديه نظام مساعدة نشك أنه موجود فى أى لغة من لغات البرمجة الأخرى. يمكن طلب هذه المساعدة بأكثر من طريقة: فى شريط القوائم هناك قائمة المساعدة Help كما فى شكل (١- ٥) ومنها يتم النقر على Product Help F1 حيث تفتح أمامك فهرس كامل لجميع موضوعات ماتلاب بلا استثناء والتى يمكنك الحصول على مساعدة لها. حاول تجربة ذلك للوصول إلى قائمة الدوال كما أشرنا فى الجزء السابق. هناك أيضا إمكانية بحث Search عن طريق كتابة أى عنوان تريد وفى الحال سيعرض لك ماتلاب كل ما يتعلق بهذا العنوان.

يمكن أيضا الحصول على مساعدة عن أى دالة مباشرة عن طريق كتابة أمر فى نافذة الأوامر كما فى المثال التالى لطلب المساعدة عن الدالة sind:

```
>> help sind
 sind   Sine of argument in degrees.
    sind(X) is the sine of the elements of X, expressed in degrees.
    For integers n, sind(n*180) is exactly zero, whereas sin(n*pi)
    reflects the accuracy of the floating point value of pi.
    Class support for input X:
       float: double, single
    See also asind, sin.
    Overloaded methods:
       distributed/sind
    Reference page in Help browser
       doc sind
```

طريقة عرض الثوابت والمتغيرات

كل مستخدم يفضل قراءة القيمة العددية للثوابت والمتغيرات بطريقة معينة. فالبعض يفضل عرض الثوابت حتى أقصى دقة فى ماتلاب وهى ١٦ خانة عشرية والبعض لا يريد العرض بهذه الدقة الزائدة التى لا يكون هناك حاجة لها فى الكثير من التطبيقات. أنظر لأمر التشكيل format التالى:

```
>> format long
>> exp(1)
ans =
  2.718281828459046
```

أمر التشكيل الطويل format long يعرض النتيجة فى ١٦ خانة عشرية كما رأينا. لاحظ أنه بعد كتابة الأمر format long ستظل طريقة العرض هذه باقية إلى أن يتم تغييرها بأمر تشكيل آخر كما يلى:

```
>> format short
>> exp(1)
ans =
   2.7183
```

حيث تم العرض فى صورة خمس خانات عشرية بدلا من ١٦. يمكن عرض النتائج فى الصورة الأسية التالية:

```
>> format short e
>> exp(1)*10
ans =
  2.7183e+001
```

هنا تم العرض فى صورة خمس خانات عشرية مضروبة فى أس حيث 10^1=e+001. هناك الكثير من طرق العرض الأخرى يمكنك مراجعتها عن طريق طلب المساعدة help format. تذكر أن دقة ماتلاب هى ١٦ خانة عشرية وعند العرض فى صورة خمس خانات عشرية فإن دقة الرقم المخزنة فى الذاكرة (١٦خانة) لا تتغير، الذى يتغير فقط هو طريقة عرض الرقم أو الثابت على الشاشة.

بعض الدوال العامة المفيدة فى ماتلاب

هناك بعض العامة التى تساعد فى تشغيل ماتلاب ولكنها لا تنفذ أى عملية حسابية منها ما يلى:

```
>> whos
 Name     Size        Bytes   Class    Attributes
 a        1x1           8     double
 ans      1x1           8     double
 b        1x1           8     double
```

```
c        1x1        8    double
x        1x1        8    double
y        1x1        8    double
z        1x1        8    double
```

حيث تعرض هذه الدالة جميع المتغيرات التى تم استخدامها فى نافذة مساحة العمل work space فى صورة جدول كما رأينا يبين اسم المتغير وحجمه حيث كما نعلم فإن كل متغير يكون عبارة عن مصفوفة، وكل المتغيرات التى تعاملنا معها حتى الآن عبارة عن مصفوفة أحادية أى لها صف واحد وعمود واحد. كذلك يعرض الجدول عدد البايتات التى يشغلها كل متغير ونوعه، وسيأتى الحديث عن أنواع المتغيرات لاحقا.

يمكن أيضا إجراء هذه الدالة على متغير واحد كما يلى:

```
>> whos x
 Name      Size           Bytes  Class     Attributes
   x        1x1               8  double
```

الدالة clear تمسح متغير أو متغيرات من الذاكرة كما يلى:

```
>> clear x
>> x
??? Undefined function or variable 'x'.
```

هنا تم مسح المتغير x من الذاكرة بالأمر clear x وبعدها تم السؤال عن نفس المتغير فكانت إجابة ماتلاب بأن هذا المتغير غير موجود. الدالة clear بدون كتابة أى متغير تمسح جميع المتغيرات الموجودة فى مساحة العمل من الذاكرة.

بعد جلسة ماتلاب طويلة تكون نافذة الأوامر مليئة بالأوامر والإجابات وفى بعض الأحيان نريد التخلص من كل هذه الكتابات لتنظيف نافذة الأوامر. الدالة clc تقوم بذلك حيث تمسح محتويات نافذة الأوامر وتضع دليل الكتابة cursor فى أول الصفحة. تذكر أن الدالة clc لا تمسح المتغيرات من الذاكرة، إنها فقط تنظف نافذة الأوامر كما يلى:

```
>>clc
>> a
a =
   12
```

حيث هنا تم تنفيذ الأمر clc فتنظفت نافذة الأوامر، وبعدها سألنا عن المتغير a فعرض لنا ماتلاب قيمة هذا المتغير مما يؤكد أن المتغير a لم يمسح من الذاكرة.

```
>> date
ans =
08-Aug-2009
```

هذه الدالة تعرض التاريخ فى صورة اليوم والشهر والسنة كما رأينا.

```
>> calendar


              Aug 2009
   S     M    Tu    W    Th    F    S
   0     0     0    0     0    0    1
   2     3     4    5     6    7    8
   9    10    11   12    13   14   15
  16    17    18   19    20   21   22
  23    24    25   26    27   28   29
  30    31     0    0     0    0    0
```

الدالة calendar تعرض نتيجة الشهر الحالى كما رأينا.

الدالة clock تعرض لك التاريخ والوقت فى متجه بالترتيب التالى من اليسار لليمين: السنة ٢٠٠٩ مثلا، ثم الشهر (شهر ٨ مثلا)، ثم اليوم، ثم الساعة، والدقيقة والثانية كما يلى:

```
>> fix(clock)
ans =
     2009      8      16      8      46      5
```

ينصح باستخدام fix قبل الأمر clock حتى يتم عرض الأرقام صحيحة بدون كسور. هنا كانت الإجابة عام ٢٠٠٩ شهر ٨ يوم ١٦ الساعة ٨ والدقيقة ٤٦ وخمس ثوان.

دوال الرسم فى ماتلاب

ماتلاب غنى جدا بدوال الرسم التى تمكنك من رسم أى دالة فى بعدين أو ثلاثة أبعاد. سنعرض هنا لبعض هذه الدوال والباقى سيأتى فى حينه فى أماكن مختلفة من الكتاب، كما أن هناك الكثير من أدوات معالجة الصور الموجودة فى مكتبة معالجة الصور.

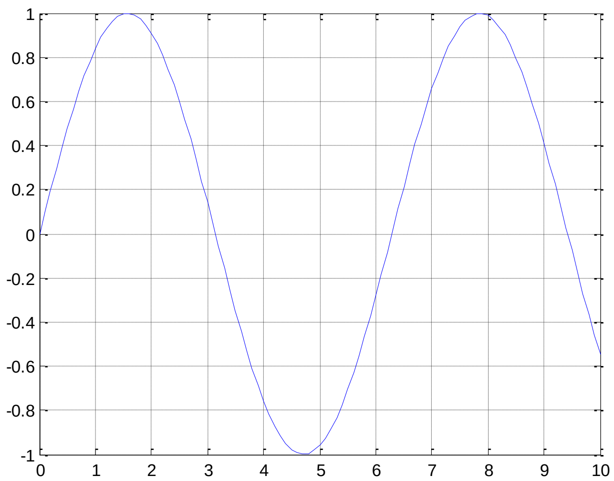
من دوال الرسم الدالة plot(x,y) التى ترسم المتغير أو المتجه y مع المتغير أو المتجه x. كمثال على ذلك سنرسم الدالة y=sin(x) كما فى الأوامر التالية وشكل (١- ٥):

```
>> x=0:0.1:10;
>> y=sin(x);
>> plot(x,y), grid
```

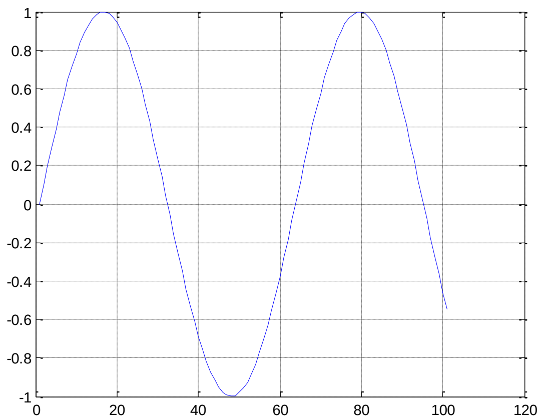
الأمر x=0:0.1:10 سيجعل x تتغير من صفر حتى 10 بالتقدير الدائرى والفرق بين كل نقطة والثانية هو 0.1، والأمر y=sin(x) سيحسب قيمة y المقابلة لكل x، ثم الأمر plot(x,y) الذى يرسم y مع x، ثم الأمر grid الذى يرسم الشبكة التى نراها فى الشكل.

دالة الرسم plot() لها متغيرات كثيرة جدا سنذكر منها ما يلى:

- الدالة plot(y) سترسم المتغير y مع فهرسه أى عند النقطة ١ و ٢ و ٣ وهكذا. لذلك فإننا لو نفذنا ذلك على الرسم السابق سنحصل على الشكل (١- ٦) حيث نلاحظ أن المحور x هنا يتغير من صفر حتى ١٢٠ حيث أن هناك ١٠٠ نقطة للمتغير x، بينما فى شكل (١- ٥) فقد تم رسم المتغير y مع قيم x التى تنتهى عند القيمة 10.



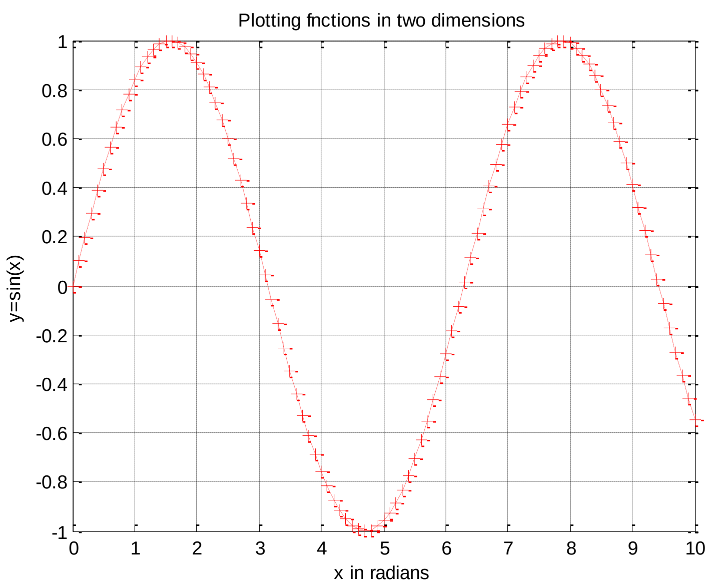
شكل (١- ٥) الدالة y=sin(x) باستخدام الأمر plot(x,y)



شكل (١- ٦) رسم الدالة y=sin(x) باستخدام الأمر plot(y)

- يمكن التحكم فى لون المنحنى وإضافة أسماء للمحاور وعنوان للشكل كما فى شكل (٧-١). فى الأمر plot( ) التالى تم إضافة علامتى تنصيص يوضع بينهما اختيار اللون بالحرف r للون الأحمر و + لتوقيع نقط الرسم و : لجعل المنحنى منقط بدلا من منحنى متصل. بعد ذلك تم إضافة مسمى للمحور x بالأمر xlabel()، وإضافة مسمى للمحور y بالأمر ylabel()، ثم إضافة عنوان للشكل بالأمر title. لاحظ أن إضافة المسميات السابقة لابد أن تكون بين علامتى تنصيص. (فى الشكل (١-٧) لون المنحنى أحمر ولكنه ربما سيظهر أسود عند طباعة الكتاب ولذلك نوصى الدارس بكتابة هذه الأوامر وملاحظة الخرج).

```
>> plot(x,y,'r+:')
>> grid
>> xlabel('x in radians')
>> ylabel('y=sin(x)')
>> title('Plotting fnctions in two dimensions')
```



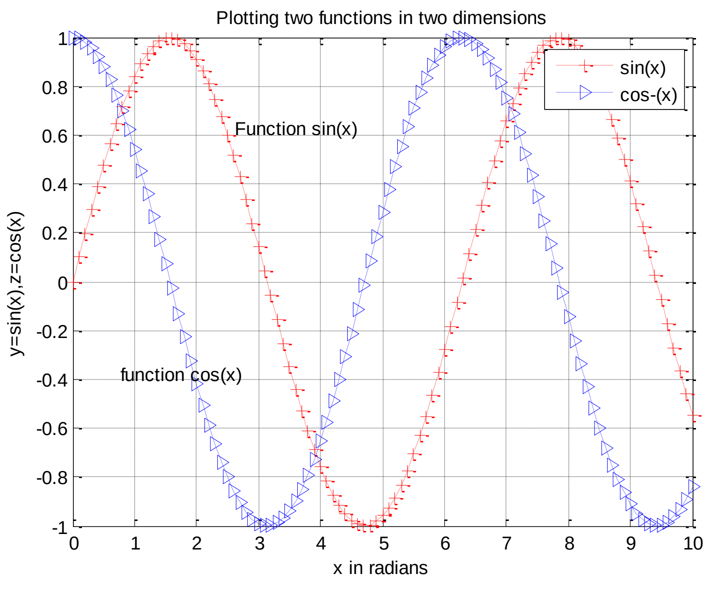
شكل (١- ٧) تسمية المحاور وإضافة عنوان للرسم

- يمكن رسم أكثر من منحنى بأمر plot() واحد الذى سنستخدمه لرسم الدالة sin(x) و cos(x) كما فى شكل (١- ٨) وكما فى الأوامر التالية:

```
>> x=0:0.1:10;
>> y=sin(x);
>> z=cos(x);
```

```
>> plot(x,y,'r+:',x,z,'b>:')
>> grid
>> title('Plotting two functions in two dimensions')
>> ylabel('y=sin(x),z=cos(x)')
>> xlabel('x in radians')
>> legend('sin(x)','cos(x)')
>> gtext('Function sin(x)')
>> gtext('function cos(x)')
```

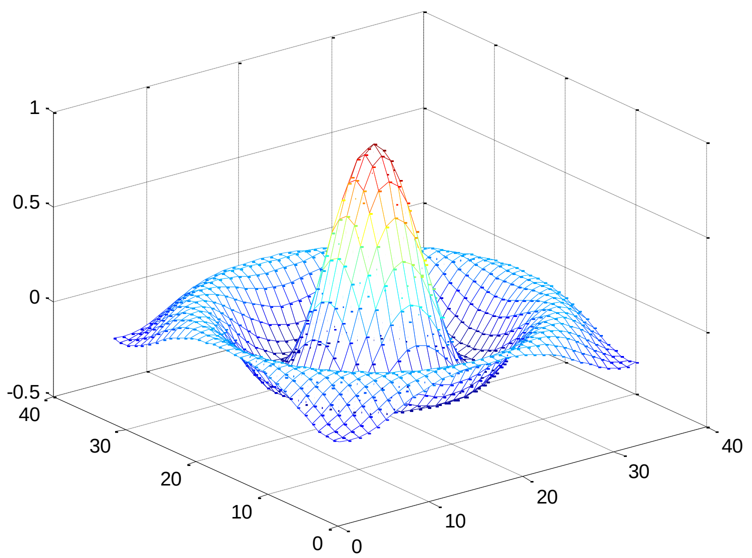
- فى هذا الشكل تم رسم منحنيين فى نفس الشكل وتم إضافة مفتاح legend لهذا الشكل نميز به كل منحنى عن الآخر كما فى الشكل حيث يوضح المفتاح أن المنحنى المنقط بعلامة زائد يمثل المنحنى sin(x)، بينما المنحنى المنقط بالمثلثات هو منحنى الدالة cos(x). هناك متغيرات كثيرة عن مفتاح الشكل من حيث مكانه وحجمه ونوع الخط فيه يمكنك معرفتها باختصار بكتابة الأمر help legend أو معرفتها بالتفصيل بالنقر على قائمة help ثم ابحث عن legend من مكان البحث حيث سيظهر لك معلومات تفصيلية بأمثلة توضيحية عن الأمر legend. لاحظ أنه بالنقر مرتين على المفتاح يمكنك تعديل النصوص المكتوبة فيه مباشرة. كما أنه يمكنك الوقوف على المفتاح والضغط عليه بالفأرة مرتين مع السحب والإسقاط فى أى مكان على الشكل لتغيير مكان المفتاح، فحاول ذلك.



شكل (١– ٨) رسم أكثر من منحنى باستخدام الأمر plot()

- يمكنك أيضا كتابة أى نص بجوار أى منحنى فى الشكل وفى أى مكان باستخدام الدالة gtext(). فى شكل (١– ٨) كتبنا النص function sin(x) والنص function cos(x) باستخدام هذه الدالة. عندما يبدأ ماتلاب بتنفيذ هذه الدالة سيظهر أمامك على الشكل خطان متقاطعان ونقطة تقاطع هذان الخطان يتحركان مع الماوس ، إختر المكان الذى تريد كتابة النص عنده، وقف بالماوس عند هذا المكان ثم انقر الماوس، ستجد أن النص الموجود فى الدالة gtext() بين علامتى التنصيص قد تم إسقاطه فى نفس المكان تماما كما فى شكل (١– ٨).

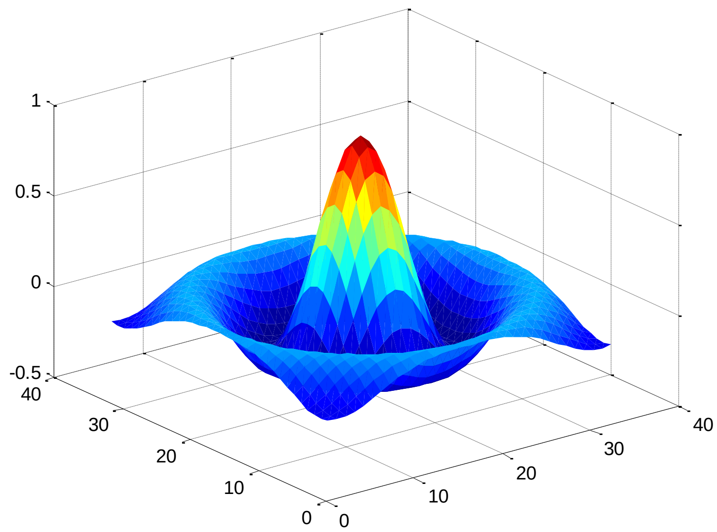
يستخدم الرسم الثلاثى الأبعاد بكثرة فى العديد من التطبيقات لإبراز الكثير من الخواص التى يصعب إظهارها فى الرسومات ثنائية الأبعاد. يحتوى ماتلاب على العديد من دوال الرسم ثلاثى الأبعاد سنقدم هنا بعضها على سبيل الأمثلة لا الحصر:



شكل (١– ٩) القبعة المكسيكية كشكل ثلاثى الأبعاد

```
>> [x y ] = meshgrid(-8 : 0.5 : 8);
>> r = sqrt(x.^2 + y.^2) + eps;
>> z = sin(r) ./ r;
>> mesh(z)
```

شكل (١– ٩) يبين القبعة المكسيكية الشهيرة وقد تم رسمها باستخدام الأربع أوامر السابقة فقط مما يبين سهولة الرسم ثلاثى الأبعاد فى ماتلاب وروعته. يمكنك الحصول على شكل أجمل من خلال إضافة ظلال لهذا الشكل الثلاثى بالأمر التالى وكما هو موضح فى شكل (١– ١٠).



شكل (١- ١٠) القبعة المكسيكية بعدإضافة ظلال لها

١-٤ ملفات البرامج (ملفات الإم m files)

إن التعامل مع ماتلاب بالطريقة التفاعلية التى رأيناها مسبقا سهلة ومباشرة وبالذات فى حالة التطبيقات القصيرة التى تتطلب سطر واحد أو حتى عدد قليل من الأسطر أو من الأوامر التى يتم كتابتها فى نافذة الأوامر command window. عندما تكون هناك حاجة إلى برنامج يتكون من العديد من الخطوات التى تريد تنفيذها ليس مرة واحدة ولكن العديد من المرات وفى كل مرة بمعاملات أو بثوابت مختلفة فإن تنفيذ مثل هذا البرنامج من نافذة الأوامر command window تكون صعبة وغير عملية. بالإضافة إلى ذلك، فإننا نحتاج دائما لكتابة البرامج فى ملفات منفصلة وتخزينها والنداء عليها للتنفيذ فى أى وقت والتعامل من خلال مساحة العمل لن يمكننا من ذلك.

يمكنك كمستخدم للماتلاب أن تفتح ملف منفصل تماما عن مساحة العمل وتكتب فيه البرنامج الخاص بك بكل خطواته وبكل التعليقات والعبارات التوضيحية التى تريدها وتخزن هذا الملف بأى إسم تريده وهذا الإسم يأخذ الامتداد m. (ومن هنا كانت تسمية هذه الملفات بملفات الإم m files). بعد أن تنتهى من تحرير هذا الملف وتخزينه يمكنك تنفيذه من مساحة العمل بكتابة إسم هذا الملف فقط واضغط على enter، حيث يتم تنفيذ البرنامج وإظهار النتائج أمامك فى مساحة العمل. لكى نوضح ذلك، من قائمة الملفات file إختر New m file (أو قد يكون هناك اختلافات بسيطة على حسب الإصدار الذى تتعامل معه) حيث ستفتح لك مساحة تحرير يمكنك فيها كتابة البرنامج الخاص بك. سنفترض أننا كتبنا برنامج الرسم

السابق الذى كتبناه فى صورة أوامر منفصلة أو خطوات فى مساحة العمل ولكننا سنكتبه هذه المرة فى مساحة التحرير فى صورة برنامج كما يلى:

```
%program to plot sin and cos functions
x=0:0.1:10; %x changes from 0 to 10 in steps of 0.1
y=sin(x);
z=cos(x);
plot(x,y,'r+:',x,z,'b>:')
grid
title('Plotting two functions in two dimensions')
ylabel('y=sin(x),z=cos(x)')
xlabel('x in radians')
legend('sin(x)','cos(x)');
```

بعد الانتهاء من كتابة البرنامج بهذا الشكل نقوم بتخزينه بأى إسم ولقد أسميناه plotting.m. بعد تخزين البرنامج بهذا الإسم نقوم بتنفيذه من مساحة العمل بكتابة إسم البرنامج فقط طما يلى:

```
>> plotting
```

حيث سيتم تنفيذ البرنامج فورا وإظهار نتيجته التى ستكون فى هذه الحالة هى الرسم الموضح فى شكل ١-٨ والذى لا داعى لإعادة رسمه مرة أخرى. البرنامج فى هذه الصورة النصية script أصبحت له عدة مميزات منها ما يلى:

١- يمكنك استدعاء البرنامج فى أى وقت وإجراء أى تعديل تريده فى أى وقت تشاء وإعادة تخزينه بعد هذا التعديل وتنفيذه من جديد.

٢- كما تلاحظ فإنه أثناء تحرير البرنامج تظهر بعض الأسطر بألوان معينه مثل السطر الأول الذى يظهر باللون الأخضر حيث يعتبر ماتلاب أى نص تتم كتابته بعد الحرف % يكون تعليقا ولا يتم تنفيذه عن طريق ماتلاب. ويمكنك كتابة الكثير من هذه التعليقات فى أى مكان فى البرنامج لتسهيل فهمه وتتبع تنفيذه.

٣- كما تلاحظ فإنه أثناء تحرير أو كتابة البرنامج فإن ماتلاب يستخدم ألوان مختلفة توضح الأمر عن المعاملات المستخدمة داخل هذا الأمر مما يساعد فى تجنب الكثير من الأخطاء.

٤- وأنت فى مساحة التحرير يمكنك تجربة تنفيذ البرنامج بالضغط على زرار تنفيذ موجود فى شريط الوظائف فى أعلى مساحة التحرير.

٥- هناك فى هذا الشريط الكثير من الأزار الأخرى التى تساعدك فى الكشف عن أخطاء البرنامج debugger بحيث يمكنك تنفيذ البرنامج خطوة بخطوة أو أوضع علامة break point يتم تنفيذ البرنامج إلى أن يقف عند هذه العلامة، وهذه وسائل تساعد كثيرا فى الكشف عن الأخطاء بالذات فى البرامج الكبيرة.

٦- تذكر دائما أنه عند تخزين البرنامج بإسم معين تجنب تماما استخدام أسماء دوال تكون سابقة التجهيز فى ماتلاب نفسه، أو كلمات مفتاحية تستخدم فى برمجته.

٧- برنامجك بهذه الصورة أصبح كما لو كان دالة من دوال ماتلاب تستدعيه للتنفيذ بمجرد كتابة إسم البرنامج فى مساحة العمل.

٨- يمكنك تحديد زمن تنفيذ البرنامج أو جزء من هذا البرنامج بوضع الكلمة المفتاحية tic عند موضع بداية حساب الزمن، والأمر t=toc عند موضع نهاية حساب الزمن. بذلك فإنه بعد تنفيذ البرنامج سيكتب لك ماتلاب ...=t وهو زمن تنفيذ هذا البرنامج أو هذا المقطع من البرنامج بالثانية. كمثال على ذلك لو وضعنا هذين الأمرين فى برنامج الرسم السابق كما يلى:

```
%program to plot sin and cos functions
tic
x=0:0.1:10; %x changes from 0 to 10 in steps of 0.1
y=sin(x);
z=cos(x);
plot(x,y,'r+:',x,z,'b>:')
grid
title('Plotting two functions in two dimensions')
ylabel('y=sin(x),z=cos(x)')
xlabel('x in radians')
legend('sin(x)','cos(x)');
t=toc
```

فإنه بتنفيذ هذا البرنامج سيظهر الرسم وسنرى فى مساحة العمل زمن التنفيذ كما يلى:

```
>> plotting
t =
  0.4863
```

١-٥ ملفات الدوال الوظيفية (ملفات الإم function m files)

ملفات الإم التى تعاملنا معها فى الجزء السابق تسمى ملفات إم نصية script m-files. معاملات أى ملف من هذه الملفات يجب أن تكون موجودة داخل الملف نفسه. نقدم هنا نوع ثانى من ملفات الإم وهو ملفات الدوال. فى هذا النوع من الملفات نريد إدخال ثوابت البرنامج كمعاملات مع الإسم عند تنفيذه. فمثلا ملف دالة لحساب مجموع ومتوسط ثلاث متغيرات سنكتبه بهذه الطريقة:

```
% Finding summation and mean of 3 numbers
function [x1,x2]=sumandmean(a,b,c);
x1=a+b+c;
x2=(a+b+c)/3;
```

من هذا البرنامج نلاحظ ما يلى:

- أول أمر فى البرنامج هو الأمر function الذى لابد أن يكون أول أمر فى البرنامج. هذا الأمر يتكون من ٤ أجزاء:

  ١- الأمر function يعقبه مسافة على الأقل.

٢- مخرجات البرنامج (أو الدالة) توضع فى صورة مصفوفة [x1,x2] حيث هذه هى النتائج التى سيخرجها البرنامج أو الدالة، ونفصل بين كل منها بفاصلة، وفى هذه الحالة ستكون المخرجات هى المجموع الذى سيوضع فى x1 والمتوسط الذى سيوضع فى x2.

٣- إسم الدالة وهو هنا sumandmean وهو أى إسم يختاره المستخدم، وهو الإسم الذى سيخزن به هذا الملف.

٤- معاملات الدالة أو مدخلاتها وهى فى حالتنا هذه ال a و b و c ولابد أن توضع بين قوسين كما رأينا ونفصل بين كل منها بفاصلة.

- لابد من تخزين البرنامج كما ذكرنا بإسم الدالة sumandmean وماتلاب بنفسه سيطلب منك ذلك عند تخزين هذا الملف.
- الآن يمكنك تنفيذ هذه الدالة من مساحة العمل عن طريق النداء عليها باستخدام قيم للمتغيرات a و b و c وهى سترد بالمجموع والمتوسط كما يلى:

```
>> [y1 y2]=sumandmean(1,2,3)
y1 =
   6
y2 =
   2
```

ومن مميزات هذه الدوال أيضا أنك يمكن النداء عليها من داخل أى برنامج آخر وتستخدم القيم المرجعة منها داخل هذا البرنامج كما لو كنت تستخدم دالة سابقة البناء داخل ماتلاب تماما مثل الدالة cos(x) التى تحسب جيب تمام الدالة x، أو الدالة log(x).

١-٦ الحسابات الرمزية فى ماتلاب

لقد تعودنا على التعامل الرقمى مع برنامج ماتلاب فمثلا يمكننا استخدام الماتلاب فى حساب قيمة دالة مثل الدالة cos(x) أو الدالة sin(x) حيث المتغير x يؤول إلى قيمة معينة بالدرجات. الجديد هنا أننا يمكننا أن نطلب من ماتلاب حساب تفاضل أو تكامل أى واحدة من هذه الدوال فيعطينا الدالة cos(x) كتفاضل للدالة sin(x) مثلا. كل هذا يتم من خلال صندوق أدوات Tool box خاص بالحسابات الرمزية ويسمى Symbolic Math Tool Box الذى سنقدم شرحا مختصرا له فى هذا الجزء.

التعامل مع المتغيرات الرمزية سيفتح مجالا واسعا وجديدا من مجالات استخدام الماتلاب وهو مجال إجراء بعض العمليات الغير رقمية على الدوال الرمزية مثل عمليات التفاضل والتكامل وغير ذلك، وهذا التطبيق مفيد جدا ليس فقط للمهندسين ولكن لكل من يتعامل مع الرياضيات، ونحن هنا سنحتاج لهذا النوع من

الحسابات فى معظم فصول هذا الكتاب وبالذات فى فصول التكامل والتفاضل الرقمى حيث سنحتاج لهذه الحسابات الرمزية لحساب التكامل أو التفاضل الحقيقى من أجل مقارنته مع ناتج التفاضل أو التكامل الرقمى، وبالذات إذا كان من الصعب الحصول على التكامل الحقيقى بصورة تحليلية مباشرة.

يستخدم صندوق أدوات الحساب الرمزى نوعا من المتغيرات الخاصة تسمى المتغيرات الرمزية symbolic variables التى ترمز للمتغير بسلسلة أحرف يتم حفظها فى هذا المتغير. المثال التالى يوضح الفرق بين متغيرات ماتلاب الحسابية العادية مثل المتغيرات متضاعفة الدقة double والمتغيرات الرمزية. الأوامر التالية تبين نسخة من نافذة الأوامر command window أو مساحة العمل work space فى ماتلاب حيث طلبنا الجذر التربيعى للرقم الحسابى ٢ بالأمر sqrt(2) فكانت الإجابة هى 1.4142 . أما عندما تم تعريف الرقم ٢ على أنه ثابت رمزى symbolic باستخدام الأمر sym(2) فكانت الإجابة هى 2^(1/2) حيث العلامة ^ تعنى الأس ، مما يعنى أن المتغير $a=\sqrt{2}$. وعلى ذلك أصبح الرمز $\sqrt{2}$ محفوظا فى المتغير a كرمز بدون حساب قيمته كما يلى:

```
>> a=sqrt(2)
a =
   1.4142
>> a=sqrt(sym(2))
a =
2^(1/2)
```

مثال آخر يوضح طريقة التعامل مع المتغيرات الرمزية سنحسب نتيجة جمع الكسرين 2/5 و 1/3 مرة على أنهم ثوابت من النوع المضاعف ومرة على أنهم ثوابت رمزية:

```
>> a=2/5+1/3
a =
   0.7333
>> a=sym(2)/sym(5)+sym(1)/sym(3)
a =
11/15
```

فى الصورة الرمزية تم جمع الكسرين بطريقة جمع الكسور الاعتيادية حيث تم توحيد مقام كل من الكسرين وحساب البسط لكل منهما ثم جمع البسطين فكانت النتيجة الكسر الاعتيادى 11/15.

يتم الإعلان عن أى متغير فى الصورة الرمزية باستخدام الأمر sym( ) كما رأينا، و sym هى اختصار لكلمة symbolic التى تعنى رمزى. الصورة العامة لهذا الأمر هى:

```
x=sym('x')
```

حيث تم وضع الرمز x بين العلامتين '' حتى يوضع هذا الرمز فى المتغير x. أنظر للمثال التالى:

```
>> a=sym('alpha')
a =
alpha
```

حيث تم وضع الرمز 'alpha' فى المتغير الرمزى a. يمكن وضع تعبير حسابى كامل مثل التعبير $\rho = \frac{1+\sqrt{5}}{2}$ كرمز فى متغير رمزى كما يلى:

```
>> rho=sym('(1+sqrt(5))/2')
rho =
5^(1/2)/2 + 1/2
>> f=rho^2-rho-1
f =
(5^(1/2)/2 + 1/2)^2 - 5^(1/2)/2 - 3/2
>> simplify(f)
ans =
0
```

لاحظ كيف تم استخدام المتغير الرمزى rho كعنصر فى تعبير كامل f وتم حساب قيمة f الرمزية ثم تبسيطها بالأمر simplify(f) الذى أعطى النتيجة صفر.

أنظر إلى معادلة الدرجة الثانية $f = ax^2 + bx + c$، هنا المتغير f يمكن وضعه فى الصورة الرمزية بالأمر f=sym('a*x^2+b*x+c') ، ولكن المتغيرات a و b و c و x فى هذا التعبير f ليست متغيرات رمزية ولذلك لا يمكن استخدامها فى عمليات حسابية رمزية مثل التفاضل والتكامل كما سنرى. لذلك فى هذه الحالة لابد من تحديد هذه المتغيرات على أنها متغيرات رمزية هى الأخرى. يمكن استخدام الأمر syms a b c x الذى يضع كل هذه المتغيرات فى الصورة الرمزية مرة واحدة بدلا من استخدام الأمر sym لكل متغير على حدة كما يلى:

```
>> syms a b c x n
>> f=x^n
f =
x^n
```

إجراء التفاضل على المتغيرات الرمزية

يمكن إجراء التفاضل على التعبيرات الرمزية بنفس قواعد التفاضل التى نعرفها تماما كما يلى:

```
>> syms x
f = sin(5*x)
f =
```

```
sin(5*x)
>> y=diff(f)
y =
5*cos(5*x)
```

كما رأينا فقد تم تعريف الدالة الرمزية f=sin(5x)، ثم طلبنا تفاضل هذه الدالة بالأمر y=diff(f) فكانت النتيجة هى y=5cos(5x). أنظر أيضا للمثال التالى:

```
>> g = exp(x)*cos(x)
g =
exp(x)*cos(x)
>> y=diff(g)
y =
exp(x)*cos(x) - exp(x)*sin(x)
```

كما يمكنك الحصول على التفاضل الثانى لأى تعبير رمزى كما يلى:

```
>> y=diff(g,2)
y =
-2*exp(x)*sin(x)
```

أو بهذه الطريقة:

```
>> y=diff(diff(g))
y =
-2*exp(x)*sin(x)
```

يمكن إجراء التفاضل على كمية ثابتة بشرط الإعلان عن هذا الثابت بالصورة الرمزية كما يلى:

```
>> c=sym('5')
c =
5
>> y=diff(c)
y =
0
```

يمكن استخدام الأمر simplify( ) لتبسيط شكل الناتج من ماتلاب إذا كان الناتج غير واضح.

يمكن إجراء التفاضل الجزئى على تعبير متعدد المتغيرات كما فى الأمثلة التالية:

```
>> syms s t;
>> f=sin(s*t);
>> y=diff(f,t)
y =
s*cos(s*t)
>> y=diff(f,s)
```

```
y =
t*cos(s*t)
```

إجراء التكامل على المتغيرات الرمزية

لإجراء التكامل على المتغيرات الرمزية فإن ماتلاب يستخدم الدالة int(f) للدوال فى متغير واحد. يمكن تحديد متغير التكامل كما فى الدالة int(f,v) التى تعطى تكامل الدالة f بالنسبة للمتغير v. إليك بعض الأمثلة على ذلك:

التكامل $\int x^2 dx$ يمكن إجراؤه كما يلى:

```
>> syms x
>> f=x^2;
>> y=int(f)
y =
x^3/3
```

يمكن أيضا إجراء التكامل المحدود كما يلى:

```
>> syms x
>> f=x^2;
>> y=int(f,0,1)
y =
1/3
```

حيث y=int(f,0,1) سيقوم بإجراء التكامل على الدالة f من 0 حتى 1.

فى حالة التعبيرات متعددة المتغيرات يمكنك تحديد المتغير الذى تريد التكامل بالنسبة له كما يلى:

```
>> syms x y n
>>f = x^n + y^n;
>> y=int(f,y)
y =
x^n*y + (y*y^n)/(n + 1)
```

حل المعادلات التفاضلية ذات المتغيرات الرمزية

يمكن استخدام ماتلاب فى حل المعادلات التفاضلية ذات المتغيرات الرمزية مثل المعادلة التفاضلية التالية:

$\frac{dy}{dt} - y + t^2 - 1 = 0$ باعتبار القيمة الابتدائية y(0)=0.5.

ماتلاب سيحل هذه المعادلة التفاضلية كما يلى:

```
>> syms y(t)
>> D2y = diff(y,2);
>> Dy = diff(y);
```

حيث هنا تم تعريف y(t) على أنه متغير رمزى، ثم تعريف D2y=diff(y,2) على أنه التفاضل الثانى للمتغير y و Dy=diff(y) على أنه التفاضل الأول للمتغير y بالنسبة للمتغير t. الآن يمكن حل المعادلة التفاضلية باستخدام الأمر dsolve( ) كما يلى:

```
>> dsolve(Dy-y+t^2-1==0, y(0)==0.5)
ans =
2*t - exp(t)/2 + t^2 + 1
```

وهذا الحل النهائى يمكن كتابته على الصورة $(t+1)^2 - 0.5e^t$.

وهذا مثال آخر لحل معادلة تفاضلية من الدرجة الثانية:

$$\frac{t^2 d^2 y}{dt^2} - \frac{t dy}{dt} - 3y = t^2 \log(t) \quad \text{حيث} \quad y(1) = -1 \quad .\frac{dy}{dt}(1) = 0$$

وسيكون حلها باستخدام ماتلاب كما يلى:

```
>> syms y(t)
>> D2y=diff(y,2);
>> Dy=diff(y);
>> dsolve(t^2*D2y-t*Dy-3*y==t^2*log(t), Dy(1)==0,y(1)==-1)
ans =
-(t^3*(log(t)/3 + 2/9) + 7/9)/t
```

بخلاف كل هذه الأمثلة فى المواضيع المختلفة فإن صندوق أدوات الحسابات الرمزية Symbolic Mathematics Tool Box يحتوى على العديد من الدوال الحسابية الأخرى والطرق المختلفة المستخدمة فى شتى أفرع الرياضيات والتى يصعب ذكرها هنا بالتفصيل وسنتركها لمن يريد الاستزادة فى هذا المجال أن يرجع إلى المساعدة التى يوفرها ماتلاب والتى تحتوى على العديد من الأمثلة المساعدة فى الكثير من التطبيقات الرياضية. ما يهمنا هنا هو هذه المواضيع التى ستهمنا فى فصول هذا الكتاب وبالذات فصول الحل الرقمى للتفاضل والتكامل والمعادلات التفاضلية حيث يمكنك استخدام الحساب الرمزى لحساب الحلول التحليلية التى سيتم حلها رقميا فى هذه الفصول.

الفصل ٢

المصفوفات والمحددات

الفصل ٢

المصفوفات والمحددات

Matrices and Determinants

٢–١ مقدمة

ربما يظن البعض أن موضوع المصفوفات والمحددات يبعد قليلا عن مواضيع التحليل العددى وربما يكون لديهم بعض الأسباب فى ذلك وهى أن معظم الموضوعات التى سنتناولها فى هذا الفصل ليست خواريزمات بالمعنى الحرفى للخواريزم ولكنها عمليات سيتم إجراؤها على المصفوفات أو المحددات مثل الجمع والطرح والمقلوب والتحليل إلى مصفوفات جانبية وغير ذلك من الكثير من العمليات التى من المفروض أن يكون أى طالب قد درسها فى السنوات الأولى من المرحلة الجامعية فى مقرر الجبر الخطى. كل هذه العمليات يمكن برمجتها على الحاسب فى صورة برامج، أو التعامل معها فى صورة دوال سابقة التجهيز فى ماتلاب بطريقة سهلة جدا وميسرة، وهذا هو أحد الأسباب الأساسية التى جعلتنا نضم هذا الفصل إلى قائمة فصول هذا الكتاب. سبب آخر لضم المصفوفات إلى موضوعات هذا الكتاب هو أن معظم طرق التحليل العددى وخواريزماته تحتاج إلى المصفوفات بطريقة أو أخرى وأقرب مثال لذلك هو الحل العددى لنظام من المعادلات الخطية. لذلك فإننا سنفترض هنا أن القارىء لديه فكرة جيدة عن التعامل مع المصفوفات ونحن نعيد هذه المواضيع سريعا جدا مع عرض كيفية التعامل معها من خلال ماتلاب لكى نوفر على القارىء مرجعا سريعا لهذه الموضوعات.

٢–٢ المصفوفات

المصفوفة هى مجموعة من الأرقام المرتبة فى عدد من الصفوف وعدد من الأعمدة والتى يمكن كتابتها فى صورتها العامة كالتالى:

$$A = \begin{bmatrix} a_{11}\ a_{12} \dots a_{1j} \dots a_{1n} \\ a_{21}\ a_{22} \dots a_{2j} \dots a_{2n} \\ \dots \\ a_{i1}\ a_{i2} \dots a_{ij} \dots a_{in} \\ \dots \\ a_{m1}\ a_{m2} \dots a_{mj} \dots a_{mn} \end{bmatrix} \qquad (٢-١)$$

حيث المصفوفة A فى المعادلة (٢– ١) تتكون من عدد m من الصفوف وعدد n من الأعمدة ولذلك فإنها تكتب عادة بالرمز A_{mn} حيث يتم كتابة عدد الصفوف m أولا ثم عدد الأعمدة n بعدها. أى عنصر من عناصر هذه المصفوفة يكتب على الصورة العامة a_{ij}، بمعنى العنصر الواقع عند تقاطع الصف i مع العمود j. إذا كانت m=n فإننا نقول أن المصفوفة A مربعة، A_{nn}. لذلك فإن المصفوفة المربعة المكونة من خمسة صفوف وخمسة أعمدة نقول عنها بأنها مصفوفة مربعة أبعادها ٥×٥.

فى أى مصفوفة مربعة تسمى العناصر a_{11} و a_{22} و a_{33} و ... و a_{nn} بعناصر القطر الرئيسى للمصفوفة، أو بطريقة أخرى فإن العناصر a_{11} و a_{22} و a_{33} و ... و a_{nn} تقع فى القطر الرئيسى للمصفوفة. مجموع عناصر هذا القطر الرئيسى لأى مصفوفة مربعة A تسمى بأثر trace هذه المصفوفة. المصفوفة التى كل عناصرها أصفار تسمى بالمصفوفة الصفرية.

المصفوفات فى ماتلاب

كلمة ماتلاب فى الأصل مأخوذة من التعبير "معمل المصفوفات Matrix Laboratory, MATLAB" وذلك لأن وحدة البيانات الأساسية فيه هى المصفوفة، (أى أن المصفوفة ذات شأن كبير فى ماتلاب الذى سيكون لغة البرمجة الأساسية فى هذا الكتاب). كل المتغيرات وكل الثوابت ينظر إليها الماتلاب على أنها مصفوفة، حتى أنك لو كتبت x=10 فإن ذلك ينظر إليه فى جميع لغات البرمجة على أن هناك متغير x وضعت فيه القيمة أو الثابت 10، ولكن فى الماتلاب يكون الوضع مختلفا حيث ينظر الماتلاب للتعبير x=10 على أن هناك متغير x وضعت فيه المصفوفة أحادية الأبعاد، أى مصفوفة من صف واحد وعمود واحد، أى مصفوفة بها عنصر واحد هو الثابت 10. هذه الطريقة الفريدة لتمثيل البيانات التى اتبعها الماتلاب سمحت بحل الكثير من المشاكل بالذات التقنية منها والتى تتعامل مع مصفوفات بيانات بأبعاد هائلة يصعب معها التعامل، أو أنها تأخذ وقتا كبيرا لحسابها ومن مثلة ذلك الصور. فتخيل مثلا أن لديك صورتين أبعاد كل منهما ١٠٢٤×١٠٢٤ بكسل وتريد ضرب هاتين الصورتين فى بعضهما، إن ذلك يتم فى الماتلاب بسهولة جدا بالمقارنة باللغات الأخرى كما سنرى.

يفصل الماتلاب بين عناصر الصف الواحد بفاصلة أو مسافة، ويفصل بين الأعمدة بالفاصلة المنقوطة. افتح ماتلاب وفى مساحة العمل نفذ العمليات التالية:

```
>> A=[1 2 3 4]
A =
    1    2    3    4
```

هنا أدخلنا لماتلاب المصفوفة A على أنها صف واحد مكون من العناصر 4 3 2 1 والتى يفصل بين كل منها مسافة (كان من الممكن أن نفصل بين كل عنصر والتالى له بفاصلة). كانت استجابة ماتلاب أنه سجل عنده مصفوفة من صف واحد وأربع أعمدة ولقد كتبناها باللون الأزرق لنميز استجابة ماتلاب عن ما يقوم المستخدم بإدخاله. لاحظ أن الدلالة على المصفوفة تكون بالقوسين المربعين. فى نفس نافذة الأوامر أكتب المصفوفة التالية وانظر كيف سيأخذها ماتلاب:

```
>> B=[1;2;3;4]
B =
    1
    2
    3
    4
```

فى المصفوفة B توجد فاصلة منقوطة تفصل بين كل عنصر والتالى له، لذلك فإن ماتلاب يعتبر كل عنصر كما لو كان صفا قائما بذاته، لذلك فقد استجاب ماتلاب بكتابة المصفوفة B فى صورة مصفوفة من أربع صفوف وعمود واحد كما رأينا. الدالة whos سابقة البناء فى ماتلاب تعرض كل المتغيرات الحية فى مساحة التشغيل مع بعض المواصفات لكل متغير كما يلى:

```
>> whos
  Name      Size            Bytes  Class     Attributes
  A         1x4                32  double
  B         4x1                32  double
```

حيث نلاحظ من استجابة ماتلاب أن A مصفوفة مكونة من صف واحد وأربع أعمدة، و B مصفوفة من أربع صفوف وعمود واحد. وعلى ذلك يمكن كتابة أى مصفوفة بأى أبعاد كالتالى:

```
>> A=[1 2 3;4 5 6;7 8 9]
A =
    1    2    3
    4    5    6
    7    8    9
```

لاحظ كيف تم الفصل بين عناصر كل صف بمسافة، والفصل بين كل عمودين بفاصلة منقوطة، فحصلنا على المصفوفة المربعة A_{3x3}. كما ذكرنا فإن المصفوفة تحدد بالقوسين المربعين، ويمكن الاستفسار عن كل عنصر على حده باستخدام الأقواس العادية كما يلى:

```
>> x=A(1,3)
x =
    3
>> x=A(2,2)
x =
    5
```

كما يمكن حساب أثر المصفوفة A السابقة كما يلى:

```
>> x=trace(A)
x =
   15
```

يوجد فى ماتلاب الكثير من المصفوفات الخاصة والتى تستخدم فى الكثير من التطبيقات وسنذكر منها على سبيل المثال وليس الحصر المصفوفتين ( )zeros و ( )ones كما يلى:

```
>> Y=zeros(4)

Y =
```

```
   0   0   0   0
   0   0   0   0
   0   0   0   0
   0   0   0   0
```

حيث zeros(n) يعطى مصفوفة مربعة nxn كل عناصرها أصفار. يمكن أيضا تحديد مصفوفة كل عناصرها وحايد وبأى أبعاد كما يلى:

```
>> N=ones(3,4)
N =
   1   1   1   1
   1   1   1   1
   1   1   1   1
```

عملية تحويل الصفوف إلى أعمدة والأعمدة إلى صفوف transpose أو بمعنى آخر دوران المصفوفة تستخدم فى الكثير من التطبيقات. هذه العملية يرمز لها بنبرة أو شرطة صغيرة بجوار إسم المصفوفة. فمثلا A' تعنى دوران المصفوفة A حول محورها. يمكن إجراء ذلك فى ماتلاب كالتالى:

```
A =
   1   2   3
   4   5   6
   7   8   9
>> B=A'
B =
   1   4   7
   2   5   8
   3   6   9
```

٢-٣ العمليات المختلفة على المصفوفات

الجمع والطرح

من المنطقى جدا أنه عند جمع أو طرح أى مصفوفتين فإن المصفوفتين يجب أن يكون لهما نفس الأبعاد. تتم عملية الجمع أو الطرح على كل عنصر من المصفوفة الأولى مع العنصر المناظر له من المصفوفة الثانية. لو قلنا مثلا أن C=A+B أى المصفوفة C تساوى حاصل جمع المصفوفة A مع المصفوفة B فإن ذلك يعنى أن كل عنصر c_{ij} فى المصفوفة C يساوى مجموع العنصرين a_{ij} و b_{ij} فى المصفوفتين A و B (أو بمعنى آخر $c_{ij}=a_{ij}+b_{ij}$ لكل قيم i و j).

```
A =
   1   2   3
```

```
    4     5     6
    7     8     9
B =
    1     4     7
    2     5     8
    3     6     9
>> C=A+B
C =
    2     6    10
    6    10    14
   10    14    18
```

بنفس الطريقة يمكن إجراء عملية الطرح.

ضرب المصفوفات

هناك ضرب المصفوفات الذى نعرفه من جبر المصفوفات والذى يرمز له بالرمز * حيث C=A*B تعنى أن المصفوفة C تساوى حاصل ضرب المصفوفة A فى المصفوفة B وأحيانا تكتب C=AB. فى هذه الحالة يكون العنصر c_{ij} على سبيل المثال فى المصفوفة C يساوى مجموع حاصل ضرب عناصر الصف i فى المصفوفة A فى عناصر العمود j فى المصفوفة B، لذلك فإن شرطا مهما جدا لصحة عملية الضرب أن يكون عدد عناصر أى صف فى المصفوفة اليسرى يساوى عدد عناصر أى عمود فى المصفوفة اليمنى. يمكن كتابة ذلك فى صورة المعادلة التالية:

$$c_{ij} = \sum_{r=1}^{n} a_{ir} b_{rj}$$

بمعنى إذا كانت C=A*B فإنه إذا كانت A أبعادها هى mxn فإن B لابد أن تكون أبعادها nxp مثلا، أى أن عدد أعمدة A ، (n)، يجب أن تساوى عدد صفوف B، (n)، وبالتالى فإن أبعاد المصفوفة C الناتجة ستكون mxp. كمثال على ذلك أنظر كيف سيعطى ماتلاب حاصل ضرب المصفوفتين A و B كالتالى:

```
A =
    1     2     3     4
    5     6     7     8
B =
    1     5
    2     6
    3     7
    4     8
>> C=A*B
```

```
C =
   30    70
   70   174
```

نتيجة مهمة من هذه الصورة للضرب هى أن A*B≠B*A.

هناك نوع آخر من ضرب المصفوفات وهو ضرب كل عنصر فى المصفوفة الأولى فى العنصر المقابل له فى المصفوفة الثانية. هذا النوع من الضرب يرمز له بالرمز التالى C=A.*B. لاحظ وجود النقطة قبل علامة الضرب (*.) لتمييز هذا النوع من الضرب عن الضرب السابق. يستخدم هذا النوع من الضرب فى خواريزمات معالجة الصور. من الطبيعى أن يكون من شرط تحقيق هذا النوع من الضرب تساوى المصفوفتين فى الأبعاد.

قسمة المصفوفات

هناك القسمة العكسية أو اليسارية left division التى تكتب على الصورة A\B وهى تكافىء حساب معكوس A ثم ضربه فى B، أى inv(A)*B. وهذا النوع من القسمة سيستخدم فى خواريزمات حل مجموعة من المعادلات الخطية كما سنرى.

هناك أيضا القسمة اليسارية على مستوى العناصر والتى تكتب على الصورة A.\B حيث يتم قسمة كل عنصر من المصفوفة B على نظيره من المصفوفة A، وبالطبع لابد من تساوى المصفوفتين فى الأبعاد. هناك أيضا قسمة يمينية على مستوى العناصر والتى تكتب على الصورة A./B حيث يتم قسمة كل عنصر فى المصفوفة A على نظيره من المصفوفة B.

٢-٤ المصفوفات المثلثة والقطرية

هناك نوعان من المصفوفات المثلثة: **المصفوفة المثلثة العلوية upper triangular** وهى المصفوفة التى يكون كل عناصرها التى تحت القطر الرئيسى تساوى أصفار، **والمصفوفة المثلثة السفلية lower triangular** التى يكون كل عناصرها التى فوق القطر الرئيسى تساوى أصفارا كما يلى:

$$A=\begin{bmatrix}a_{11} & a_{12} & a_{13}\\ 0 & a_{22} & a_{23}\\ 0 & 0 & a_{33}\end{bmatrix}\qquad B=\begin{bmatrix}a_{11} & 0 & 0\\ a_{21} & a_{22} & 0\\ a_{31} & a_{32} & a_{33}\end{bmatrix}\qquad C=\begin{bmatrix}a_{11} & 0 & 0\\ 0 & a_{22} & 0\\ 0 & 0 & a_{33}\end{bmatrix}$$

حيث المصفوفة A تعتبر مصفوفة مثلثة علوية، والمصفوفة B تعتبر مصفوفة مثلثة سفلية. أما إذا كانت كل عناصر المصفوفة تساوى أصفار ما عدا عناصر القطر الرئيسى مثل المصفوفة C، فإن هذه المصفوفة تسمة مصفوفة قطرية diagonal.

المصفوفة المربعة القطرية التى كل عناصر قطرها الأساسى تساوى وحايد تسمى **مصفوفة الوحدة identity matrix**. عادة تأخذ مصفوفة الوحدة الرمز I_{nxn} حيث n هى أبعاد تلك المصفوفة كما يلى:

$$I_{1x1} = [1] \quad ، I_{2x2} = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix} \quad ، I_{3x3} = \begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix} \quad ،\ldots \quad ، I_{nxn} = \begin{bmatrix}1 & 0 & 0 & \ldots & 0\\0 & 1 & 0 & \ldots & 0\\ & & \ldots\ldots\ldots & & \\0 & 0 & 0 & \ldots & 1\end{bmatrix}$$

المصفوفة I تسمى مصفوفة الوحدة لأنها عند ضربها فى أى مصفوفة A تعطى نفس المصفوفة كما يلى:

$$I_{mxm} * A_{mxn} = A_{mxn} * I_{nxn} = A_{mxn}$$

الدالة eye(n) فى ماتلاب تعطى مصفوفة وحدة مربعة بعدها n كما يلى:

```
>> B=eye(4)
B =
    1    0    0    0
    0    1    0    0
    0    0    1    0
    0    0    0    1
```

المصفوفة المتماثلة symmetrical matrix هى المصفوفة التى عند دورانها تعطى نفس المصفوفة، أى $A = A'$.

معكوس المصفوفة

معكوس المصفوفة inverse of a matrix هو مصفوفة عند ضربها فى المصفوفة الأساسية تعطى مصفوفة الوحدة. بمعنى آخر، إذا افترضنا أن المصفوفة B هى معكوس المصفوفة A (وتكتب $B=A^{-1}$) فإن B*A=A*B=I. أحد شروط تواجد المعكوس لأى مصفوفة هو أن تكون محددة هذه المصفوفة لا تساوى الصفر، ونقول فى هذه الحالة أن المصفوفة قابلة للعكس invertible.

يوجد فى ماتلاب الدالة inv( ) سابقة البناء التى تحسب معكوس أى مصفوفة مباشرة كما يلى:

```
A =
   10    2    3
    4   20    3
    3    3   10
>> B=inv(A)
B =
   0.1121   -0.0065   -0.0317
  -0.0182    0.0534   -0.0106
  -0.0282   -0.0141    0.1127
>> A*B
ans =
   1.0000         0   -0.0000
  -0.0000    1.0000         0
        0         0    1.0000
```

٢–٥ المحددات

يرمز للمحددة المصاحبة لأى مصفوفة مربعة A بالرمز det A أو det(A) أو |A| والرمز الأخير هو الأكثر استخداما.

المحددات عبارة هدف حسابى (ثابت) مفيد فى الكثير من التطبيقات وبالذات فى حل مجموعة من المعادلات الخطية.

بالنسبة لأى مصفوفة مربعة A_{2x2} يمكن كتابة محددها كما يلى:

بفرض $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$

فإن $\det(A) = |A| = a_{11}a_{22} - a_{21}a_{12}$

بالنسبة للمصفوفة المربعة الثلاثية A_{3x3} يمكن حساب محددها عن طريق تكرار أول عمودين كما يلى:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{11} & a_{12} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} \end{bmatrix}$$

يتم حساب المحددة |A| لهذه المصفوفة الثلاثية عن طريق مجموع حاصل ضرب العناصر التى تمر عليها الأسهم الزرقاء مطروحا منه مجموع مضروب العناصر التى تمر عليها الأسهم الحمراء. (لاحظ أننا قمنا بتكرار أول صفين على يسار المصفوفة). وعلى ذلك يمكن كتابة المحددة |A| كما يلى:

$$|A| = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{31}a_{22}a_{13} - a_{32}a_{23}a_{11} - a_{33}a_{22}a_{12}$$

كمثال على ذلك سنحسب قيمة محدد المصفوفة الثلاثية التالية:

$A = \begin{bmatrix} 2 & 3 & 5 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}$ بتكرار أول عمودين على يمين المصفوفة تصبح كما يلى: $A = \begin{bmatrix} 2 & 3 & 5 & 2 & 3 \\ 1 & 0 & 1 & 1 & 0 \\ 2 & 1 & 0 & 2 & 1 \end{bmatrix}$

وبالتالى يمكن كتابة المحددة |A| كما يلى:

$$|A| = (2x0x0) + (3x1x2) + (5x1x1) - (2x0x5) - (1x1x2) - (0x1x3)$$
$$= 0 + 6 + 5 - 0 - 2 - 0 = 9$$

تذكر أن هذه الطريقة تعمل مع المحددات الثنائية والثلاثية فقط. يوجد فى ماتلاب دالة det( ) تعطى قيمة المحدد لأى مصفوفة مربعة كما يلى:

```
A =
    2    3    5
    1    0    1
    2    1    0
>> x=det(A)
x =
    9
```

محددة أى مصفوفة مثلثة علوية تساوى حاصل ضرب عناصر قطرها الأساسى كما يلى:

```
A =
    2    7   -3    8    3
```

```
  0   -3    7    5    1
  0    0    6    7    6
  0    0    0    9    8
  0    0    0    0    4
>> x=det(A)
x =
    -1296
```

لاحظ أن نفس النتيجة يمكن الحصول عليها من ضرب عناصر القطر الرئيسى كما يلى:

$$|A| = 2x(-3)x6\ x9x4 = -1296$$

المحددات الثانوية والمساعدة minors and cofactors

المحددة الثانوية minor هى المحددة المتبقية من أى مصفوفة مربعة ذات الأبعاد nxn بعد حذف الصف والعمود المشتركين لأحد عناصر هذه المصفوفة. أبعاد هذه المحددة المتبقية ستكون (n-1)x(n-1)، وهذه المحددة الثانوية يرمز لها عادة بالرمز $|M_{ij}|$.

المحددة المساعدة cofactor هى المحددة الثانوية بعد إعطائها إشارة على النحو التالى: $(-1)^{i+j}|M_{ij}|$.

من أمثلة ذلك افترض المصفوفة الثلاثية التالية:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

المحددة الثانوية $|M_{11}|$ تأتى من حذف الصف الأول والعمود الأول كالتالى:

$$|M_{11}| = \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix}$$ والمحددة المساعدة المقابلة ستكون $$\alpha_{11} = (-1)^{1+1}|M_{11}|$$

والمحددة الثانوية $|M_{12}|$ تأتى من حذف الصف الأول والعمود الثانى كالتالى:

$$|M_{12}| = \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix}$$ والمحددة المساعدة المقابلة ستكون $$\alpha_{12} = (-1)^{1+2}|M_{12}|$$

وبالمثل المحددة الثانوية $|M_{13}|$ تأتى من حذف الصف الأول والعمود الثالث كالتالى:

$$|M_{13}| = \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}$$ والمحددة المساعدة المقابلة ستكون $$\alpha_{13} = (-1)^{1+3}|M_{12}|$$

وبنفس الطريقة يمكن حساب باقى المحددات الثانوية والمساعدة.

ولنفترض المثال العددى التالى:

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 2 & -4 & 2 \\ -1 & 2 & -6 \end{bmatrix}$$

$$|M_{11}| = \begin{vmatrix} -4 & 2 \\ 2 & -6 \end{vmatrix} = 20$$ و $$\alpha_{11} = (-1)^{1+1}20 = 20$$

$$|M_{12}| = \begin{vmatrix} 2 & 2 \\ -1 & -6 \end{vmatrix} = -10 \quad \text{و} \quad \alpha_{12} = (-1)^{1+2}(-10) = 10$$
$$|M_{13}| = \begin{vmatrix} 2 & -4 \\ -1 & 2 \end{vmatrix} = 0 \quad \text{و} \quad \alpha_{13} = (-1)^{1+3}(0) = 0$$

وهكذا باقى المحددات الثانوية والمساعدة.

قيمة المحددة لأى مصفوفة تساوى مجموع حاصل ضرب عناصر أى صف أو عمود فى المصفوفة المساعدة لكل عنصر. فمثلا محددة أى مصفوفة ثلاثية A يمكن كتابتها كالتالى:

$$|A| = a_{11}\alpha_{11} + a_{12}\alpha_{12} + a_{13}\alpha_{13}$$

وبتطبيق ذلك على المصفوفة الثلاثية السابقة نحصل على:

$$|A| = 1x20 + 2x10 + (-3)x0 = 40$$

وباستخدام ماتلاب لحساب محددة هذه المصفوفة نحصل على:

```
A =
    1    2   -3
    2   -4    2
   -1    2   -6
>> x=det(A)
x =
   40
```

هذه الطريقة هى المستخدمة فى حساب محددات المصفوفات ذات الدرجة الرابعة أو أكثر. وعلى ذلك فإذا كانت كل عناصر أى صف أو عمود فى أى مصفوفة تساوى أصفارا، فإن محددة هذه المصفوفة ستساوى صفرا. أيضا إذا كانت عناصر أى صف أو عمود فى أى مصفوفة تساوى ثابت مضروبا فى عناصر أى صف أو عمود آخر، فإن محددة هذه المصفوفة ستساوى صفرا أيضا.

٢-٦ قانون كرامر Cramer's rule

يستخدم هذا القانون فى حل مجموعة من المعادلات الخطية باستخدام المصفوفات والمحددات. إفترض أنه مطلوب حل مجموعة المعادلات الخطية التالية:

$$a_{11}x + a_{12}y + a_{13}z = A$$
$$a_{21}x + a_{22}y + a_{23}z = B$$
$$a_{31}x + a_{32}y + a_{33}z = C$$

من هذه المجموعة من المعادلات يمكن حساب المحددات التالية:

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \quad \text{و} \quad D_1 = \begin{vmatrix} A & a_{12} & a_{13} \\ B & a_{22} & a_{23} \\ C & a_{32} & a_{33} \end{vmatrix} \quad \text{و} \quad D_2 = \begin{vmatrix} a_{11} & A & a_{13} \\ a_{21} & B & a_{23} \\ a_{31} & C & a_{33} \end{vmatrix} \quad D_3 = \begin{vmatrix} a_{11} & a_{12} & A \\ a_{21} & a_{22} & B \\ a_{31} & a_{32} & C \end{vmatrix}$$

تبعا لقانون كرامر فإن المجاهيل x و y و z يمكن كتابتها من هذه المحددات كما يلى:

$$z = D_3/\Delta \quad \text{و} \quad y = D_2/\Delta \quad \text{و} \quad x = D_1/\Delta$$

بالطبع لابد أن تكون المصفوفة Δ لا تساوى صفر لأنها فى المقام فى معادلات المجاهيل x و y و z. أى مصفوفة تكون محددتها تساوى صفر تسمى مصفوفة فريدة singular. وبالتالى فإن أى نظام من المعادلات الخطية تكون مصفوفة الثوابت به فريدة لن يكون له حل، والمصفوفات الفريدة سيأتى ذكرها كثيرا عند التعامل مع خوارزمات التحليل العددى.

كمثال على ذلك دعنا نحل المعادلات الثلاث الخطية التالية:

$$\begin{aligned} 2x - y + 3z &= 5 \\ -4x - 3y - 2z &= 8 \\ 3x + y - z &= 4 \end{aligned}$$

باستخدام ماتلاب فى حساب المحددات الأربعة سنحصل على:

$$D_2 = \begin{vmatrix} 2 & 5 & 3 \\ -4 & 8 & -2 \\ 3 & 4 & -1 \end{vmatrix} = -170 \;\text{و}\; D_1 = \begin{vmatrix} 5 & -1 & 3 \\ 8 & -3 & -2 \\ 4 & 1 & -1 \end{vmatrix} = 85 \;\text{و}\; \Delta = \begin{vmatrix} 2 & -1 & 3 \\ -4 & -3 & -2 \\ 3 & 1 & -1 \end{vmatrix} = 35$$

$$D_3 = \begin{vmatrix} 2 & -1 & 5 \\ -4 & -3 & 8 \\ 3 & 1 & 4 \end{vmatrix} = -55$$

وعلى ذلك فإن: x=85/35=2.4286 و y=-170/35=-4.8571 و z=-55/35=-1.5714

٢-٧ حل مجموعة من المعادلات الخطية باستخدام تحليل مصفوفة المعاملات إلى مصفوفتين مثلثتين، علوية وسفلية

إفترض أننا نريد حل مجموعة المعادلات الخطية التالية:

Ax=b (٢-٢)

حيث A هى مصفوفة المعاملات، و x هو متجه المجاهيل، و b هو متجه الثوابت.

بتحليل مصفوفة المعاملات A إلى مصفوفتين الأولى مصفوفة مثلثة علوية والثانية مصفوفة مثلثة سفلية فإنه يمكن كتابة المعادلة (٢- ٢) كما يلى:

LUx=b (٣-٢)

حيث A=LU و U هى مصفوفة مثلثة علوية، و L هى مصفوفة مثلثة سفلية. فى المعادلة (٢- ٣) بوضع Ux=y يمكن إعادة كتابتها كما يلى:

Ly=b (٤-٢)

المعادلة (٢– ٤) يمكن حلها بالتعويض المباشر لإيجاد متجه المجاهيل y. بمعلومية متجه المجاهيل y يمكن التعويض فى المعادلة Ux=y لإيجاد متجه المجاهيل x. الجديد أو التبسيط هنا هو أنه بتحليل المصفوفة A إلى المصفوفتين U و L فإن حساب كل من y و x تصبح عملية تعويض حسابى مباشر كما سنرى فى المثال التالى.

مثال ٢–١

إفترض مجموعة المعادلات الخطية Ax=b التالية:

$$\begin{bmatrix} 2 & 6 & 2 \\ -3 & -8 & 0 \\ 4 & 9 & 2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} \qquad (٢–٥)$$

سنفترض أن مصفوفة المعاملات A يمكن تحليلها إلى مصفوفتين مثلثتين علوية وسفلية (سنرى ذلك بعد قليل) كما يلى:

$$\underset{L}{\begin{bmatrix} 2 & 0 & 0 \\ -3 & 1 & 0 \\ 4 & -3 & 7 \end{bmatrix}}\underset{U}{\begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix}}\underset{x}{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}=\underset{b}{\begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix}} \qquad (٢–٦)$$

بوضع Ux=y فى المعادلة (٢– ٦) يمكن إعادة كتابتها كما يلى:

$$\begin{bmatrix} 2 & 0 & 0 \\ -3 & 1 & 0 \\ 4 & -3 & 7 \end{bmatrix}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}=\begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix} \qquad (٢–٧)$$

من هذه المعادلة نجد أن $2y_1=2$ ومنها $\mathbf{y_1=1}$. وأن $-3y_1+y_2=2$ ومنها $\mathbf{y_2=5}$، و $4y_1-3y_2+7y_3=3$ ومنها $\mathbf{y_3=2}$.

بمعرفة المتجه y يمكن التعويض فى المعادلة Ux=y لنوجد قيم المتجه x كما يلى:

$$\begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 1 \\ 5 \\ 2 \end{bmatrix} \qquad (٢–٨)$$

من المعادلة (٢–٨) سنجد أن $\mathbf{x_3=2}$ و $x_2+3x_3=5$ ومنها $\mathbf{x_2=-1}$ و $x_1+3x_2+x_3=1$ ومنها $\mathbf{x_1=2}$.

إذن يبقى السؤال هو كيف سنضع مصفوفة المعاملات A فى صورة مصفوفتين مثلثتين علوية وسفلية؟

المصفوفات الأولية

هناك ثلاث من العمليات الأولية يمكن إجراؤها على صفوف أى مصفوفة:

١– ضرب أحد صفوف المصفوفة فى كمية ثابتة.

٢– استبدال صفين بحيث يوضع كل منهما مكان الآخر.

٣– إضافة ثابت مضروب فى أحد الصفوف إلى صف آخر.

المصفوفة الأولية: هى مصفوفة وحدة تم عليها إجراء واحدة من العمليات الأولية السابقة.

مثال ٢-٢

سنقدم هنا أمثلة على بعض المصفوفات الأولية:

المصفوفة الأولية: $A=\begin{bmatrix}1 & 0\\ 0 & -3\end{bmatrix}$ هى مصفوفة الوحدة $I_{2x2}=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$ مع ضرب صفها الثانى فى 3-.

معكوس المصفوفة الأولية الناتجة من ضرب أحد صفوف مصفوفة الوحدة فى ثابت m هو نفس المصفوفة الأولية مع وضع الثابت يساوى 1/m بدلا من m. فمثلا معكوس المصفوفة A السابقة سيكون $A^{-1}=\begin{bmatrix}1 & 0\\ 0 & -1/3\end{bmatrix}$ ويمكنك إثبات ذلك بإجراء AA^{-1}.

المصفوفة الأولية: $A=\begin{bmatrix}1&0&0&0\\0&0&0&1\\0&0&1&0\\0&1&0&0\end{bmatrix}$ هى مصفوفة الوحدة $I_{4x4}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$ مع استبدال الصفين الثانى والرابع. معكوس المصفوفة الأولية الناتجة من استبدال أى صفين مع بعضهما البعض يكون هو نفس المصفوفة الأولية. أى أن معكوس المصفوفة A السابقة سيكون هو نفس المصفوفة A، أى أن $A^{-1}=A$.

المصفوفة الأولية: $A=\begin{bmatrix}1&0&3\\0&1&0\\0&0&1\end{bmatrix}$ هى مصفوفة الوحدة $I_{3x3}=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}$ مع ضرب الصف الثالث فى 3 وجمعه على الصف الأول. معكوس المصفوفة الأولية الناتجة من ضرب أحد صفوف مصفوفة الوحدة فى ثابت m وجمعه مع صف آخر يكون هو نفس المصفوفة الأولية الناتجة مع وضع m=-m. أى أن معكوس المصفوفة A السابقة سيكون $A=\begin{bmatrix}1&0&-3\\0&1&0\\0&0&1\end{bmatrix}$.

كنتيجة لكل ذلك فإن كل مصفوفة أولية تكون قابلة للعكس، ومعكوسها هو مصفوفة أولية أيضا.

يمكن تحويل أى مصفوفة مربعة A إلى مصفوفة مثلثة علوية عن طريق ضرب المصفوفة A من اليسار بتتابع من المصفوفات الأولية كما يلى:

$$E_kE_{k-1}\dots E_2E_1A=U \qquad (٢-٩)$$

حيث المصفوفات E_1 و E_2 و .... حتى E_k كلها عبارة عن مصفوفات أولية. بضرب طرفى المعادلة (٢- ٩) من اليسار فى معكوس كل مصفوفة من المصفوفات الأولية، مع العلم أن كل مصفوفة أولية تكون قابلة للعكس ومعكوسها هو مصفوفة أولية أيضا كما ذكرنا.

$$E_1^{-1}E_2^{-1}\dots E_{k-1}^{-1}E_k^{-1}E_kE_{k-1}\dots E_2E_1A=E_1^{-1}E_2^{-1}\dots E_{k-1}^{-1}E_k^{-1}U \qquad (٢-١٠)$$

المعادلة (٢- ١٠) يمكن كتابتها على الصورة التالية:

$$A=E_1^{-1}E_2^{-1}\dots E_{k-1}^{-1}E_k^{-1}U \qquad (٢-١١)$$

وبالتالى فإن المعادلة (٢- ١١) يمكن كتابتها على الصورة التالية:

$$A=LU \qquad (٢-١٢)$$

حيث: $L = E_1^{-1}E_2^{-1} \dots E_{k-1}^{-1}E_k^{-1}$

وبذلك أمكن تحليل المصفوفة A إلى مصفوفتين مثلثتين علوية وسفلية على الصورة A=LU. المثال التالى سيوضح ذلك بالتفصيل.

مثال ٢–٣

سنتبع الخطوات السابقة فى تحليل المصفوفة $A = \begin{bmatrix} 2 & 6 & 2 \\ -3 & -8 & 0 \\ 4 & 9 & 2 \end{bmatrix}$ إلى الصورة A=LU.

حيث أن A عبارة عن مصفوفة ثلاثية الأبعاد فإن مصفوفة الوحدة التى سنستخدمها فى كل الخطوات التالية ستكون مصفوفة الوحدة $I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

١– نجعل العنصر القطرى $a_{11}=1$. بضرب الصف الأول من المصفوفة I فى ½ نحصل على أول مصفوفة أولية $E_1 = \begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. معكوس المصفوفة E_1 هو $E_1^{-1} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. بضرب المصفوفة A فى المصفوفة الأولية E_1 نحصل على:

$$U_1 = E_1A = \begin{bmatrix} 1 & 3 & 1 \\ -3 & -8 & 0 \\ 4 & 9 & 2 \end{bmatrix}$$

٢– نجعل العنصر $a_{21}=0$ فى المصفوفة السابقة، U_1. بضرب الصف الأول فى مصفوفة الوحدة I فى 3 وجمعه على الصف الثانى نحصل على المصفوفة الأولية الثانية: $E_2 = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. معكوس المصفوفة E_2 هو $E_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

وبضرب المصفوفة U_1 فى E_2 نحصل على: $U_2 = E_2U_1 = \begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 4 & 9 & 2 \end{bmatrix}$. بذلك نكون قد صفرنا a_{21} وبالصدفة أصبح $a_{22}=1$ أيضا، لذلك سننتقل إلى الصف الثالث.

٣– تصفير a_{31} فى المصفوفة U_2. بضرب الصف الأول من مصفوفة الوحدة I فى 4- والجمع على الصف الثالث نحصل على المصفوفة الأولية الثالثة: $E_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -4 & 0 & 1 \end{bmatrix}$ ، معكوس المصفوفة E_3 هو: $E_3^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 4 & 0 & 1 \end{bmatrix}$، وبضرب المصفوفة U_2 فى E_3 نحصل على: $U_3 = E_3U_2 = \begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 0 & -3 & -2 \end{bmatrix}$.

٤- تصفير العنصر a_{32} فى المصفوفة U_3. بضرب الصف الثانى من مصفوفة الوحدة I فى 3 والجمع على الصف الثالث نحصل على المصفوفة الأولية الرابعة: $E_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}$ ، معكوس المصفوفة E_4 هو: $E_4^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix}$،

وبضرب المصفوفة U_3 فى E_4 نحصل على: $U_4 = E_4 U_3 = \begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 7 \end{bmatrix}$.

٥- نجعل العنصر $a_{33}=1$ فى المصفوفة U_4. بضرب الصف الثالث من مصفوفة الوحدة I فى 1/7 على المصفوفة الأولية الخامسة: $E_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/7 \end{bmatrix}$ ، معكوس المصفوفة E_5 هو: $E_5^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 7 \end{bmatrix}$، وبضرب المصفوفة U_4 فى E_5 نحصل على: $U_5 = E_5 U_4 = \begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} = U$.

٦- بذلك تكون U_5 هى المصفوفة المثلثة العلوية النهائية، وتكون المصفوفة المثلثة السفلية L هى:

$$L = E_1^{-1}E_2^{-1}E_3^{-1}E_4^{-1}E_5^{-1} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 4 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 7 \end{bmatrix}$$

بإجراء عملية الضرب السابقة نحصل على $L = \begin{bmatrix} 2 & 0 & 0 \\ -3 & 1 & 0 \\ 4 & -3 & 7 \end{bmatrix}$

وبذلك يمكن كتابة المصفوفة A كما يلى:

$$A = LU = \begin{bmatrix} 2 & 0 & 0 \\ -3 & 1 & 0 \\ 4 & -3 & 7 \end{bmatrix}\begin{bmatrix} 1 & 3 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

وبذلك يمكن استخدام هذا التحليل فى حل نظم المعادلات الخطية كما أشرنا مسبقا.

جدير بالذكر أن تحليل أى مصفوفة إلى مصفوفتين مثلثتين علوية وسفلية ليست عملية فريدة، بمعنى أنه يمكن الحصول على أكثر من مصفوفة مثلثة علوية وأكثر من مصفوفة مثلثة سفلية. برنامج ماتلاب به الدالة lu( ) سابقة التجهيز التى تقوم بتحليل أى مصفوفة إلى علوية وسفلية، وبتنفيذ هذه الدالة على الدالة A التى فى المثال السابق سنحصل على إجابة مختلفة كما يلى:

```
A =
    2    6    2
   -3   -8    0
    4    9    2
>> [L U]=lu(A)
L =
    0.5000    1.0000         0
   -0.7500   -0.8333    1.0000
    1.0000         0         0
U =
```

```
4.0000    9.0000    2.0000
     0    1.5000    1.0000
     0         0    2.3333
```

وللتأكد من صحة هذا التحليل سنحسب B=L*U كما يلى:

```
>> B=L*U
B =
    2    6    2
   -3   -8    0
    4    9    2
```

مما يدل على أن عملية التحليل تلك ليست عملية فريدة.

٢–٨ تمارين

بالنسبة للتمارين ١ و ٢ و ٣و ٤ المصفوفات A و B و C و D هى كما يلى:

$$A=\begin{bmatrix}1&-1&-4\\5&7&-2\\3&-5&6\end{bmatrix} \text{ و } B=\begin{bmatrix}5&9&-3\\-2&8&2\\7&-4&6\end{bmatrix} \text{ و } C=\begin{bmatrix}4&6\\-3&8\\5&-2\end{bmatrix} \text{ و } D=\begin{bmatrix}1&-2&3\\-3&6&-4\end{bmatrix}$$

١– نفذ العمليات التالية إذا كانت ممكنة وتحقق من إجابتك باستخدام ماتلاب:

أ– A+B

ب– A+C

ت– B+D

ث– C+D

ج– A-C

ح– B-D

خ– C-D

٢– أعد تمرين ١ على العمليات التالية:

أ– AB

ب– AC

ت– BD

ث– CD

ج– BA

ح– CA

خ- DA

د- DC

٣- أعد تمرين ١ على العمليات التالية:

أ- det(A)

ب- det(B)

ت- det(c)

ث- det(d)

ج- det(AB)

ح- det(AC)

٤- استخدم قانون كرامر لحل مجموعة المعادلات الخطية التالية:

$$\begin{aligned} x - 2y + z &= -4 \\ -2x + 3y + z &= 9 \\ 3x + 4y - 5z &= 0 \end{aligned}$$

٥- استخدم قانون كرامر لحل مجموعة المعادلات الخطية التالية، استخدم ماتلاب لحساب المحددات:

$$\begin{bmatrix} 2 & 4 & 3 & -2 \\ 2 & -4 & 1 & 3 \\ -1 & 3 & -4 & 2 \\ 2 & -2 & 2 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} 1 \\ 10 \\ -14 \\ 7 \end{bmatrix}$$

٦- استخدم التحليل LU لحل نظم المعادلات الخطية التالية:

أ-

$$\begin{aligned} 3x_1 - 6x_2 &= 0 \\ -2x_1 + 5x_2 &= 1 \end{aligned}$$

ب-

$$\begin{aligned} 3x_1 - 6x_2 - 3x_3 &= -3 \\ 2x_1 \qquad + 6x_3 &= -22 \\ -4x_1 + 7x_2 + 4x_3 &= 3 \end{aligned}$$

٧- استخدم التحليل LU لحل نظم المعادلات الخطية التالية:

أ-

$$\begin{bmatrix} 2 & -2 & -2 \\ 0 & -2 & 2 \\ -1 & 5 & 2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -4 \\ -2 \\ 6 \end{bmatrix}$$

ب-

$$\begin{bmatrix} -3 & 12 & -6 \\ 1 & -2 & 2 \\ 0 & 1 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -33 \\ 7 \\ -1 \end{bmatrix}$$

الفصل ٣

مقدمة عن التحليل العددى

الفصل ٣

مقدمة عن التحليل العددى

٣-١ مقدمة

لقد ذكرنا من قبل أن الهدف من التحليل العددى هو تقديم خوارزمات تؤدى إلى حلا للمسائل الصعبة التى لا يوجد لها حلا بالطرق التحليلية ويتم ذلك باستخدام الحاسب. من الممكن أيضا النظر للحل العددى لأى مشكلة على أنه خطوة من خطوات محاكاة هذه المشكلة على الحاسب وهذه المحاكاة تعتبر ميزة عظيمة فى دراسة أداء الحل المقدم تحت الظروف والأحوال المختلفة لمعرفة هل هذا الحل مستقر أم أنه يتغير بتغير الظروف المحيطة كما أن هذه المحاكاة تمكننا من دراسة أداء الحل فى ظروف يصعب فرضها فى الوضع الطبيعى حيث قد تحتاج إلى تكلفة عالية ووقت كبير. استخدام الحاسب فى حل المشاكل بالتحليل العددى ليس خاليا من الصعوبات أو العيوب التى يجب تجنبها والحرص من الوقوع فيها نتيجة أخطاء حتمية تصاحب استخدام الحاسب فى حل المشاكل الكبيرة التى قد تكون حساسة لهذه الأخطاء التى تبدو لنا بسيطة ولكنها فى ظروف معينة قد تؤدى إلى كوارث. سنقدم فى هذا الفصل مراجعة عن ميزة استخدام الحاسب فى التحليل العددى والمحاكاة وسنقدم لبعض من أخطاء الحاسب التى تبدو لنا يسيطة كما ذكرنا ولكنها أدت بالفعل إلى كوارث.

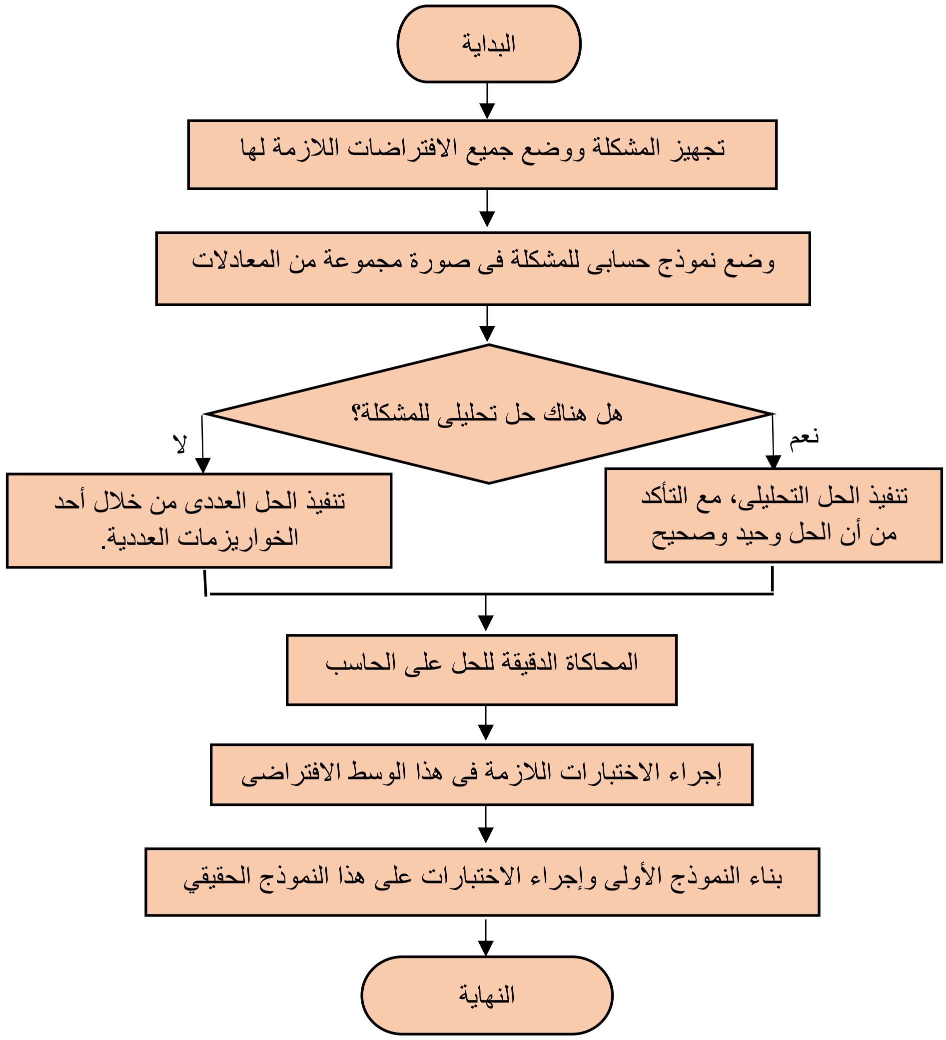
٣-٢ استخدام الحاسب فى المحاكاة والتحليل العددى

شكل ٣-١ يبين مخطط صندوقى لتسلسل خطوات حل أى مشكلة أو مشروع صغيرا أو كبيرا حتى نكون أكثر عمومية. يبدأ الحل كما فى الشكل بتحديد المشكلة من حيث عدد متغيراتها ومدخلاتها ومخرجاتها المتوقعة والحدود التى لا يجب أن يخرج عنها الحل والطرق المختلفة المقترحة للتحقق من صحة حل هذه المشكلة فى حالة الوصول إلى ذلك. الخطوة التالية لتحديد المشكلة ستكون وضع نموذج حسابى للمشكلة. هذا النموذج يكون فى الغالب فى صورة عدد من المعادلات فى عدد من المجاهيل. عدد هذه المعادلات والمجاهيل سيتوقف على طبيعة المشكلة المراد حلها، فقد يصل هذا العدد إلى آلاف من المعادلات التى قد تكون خطية أو غير خطية وقد تكون تفاضلية أو تكاملية. تخيل مثلا عمل نموذج لسيارة أو طائرة أو كوبرى أو دائرة كهربية معقدة.

يأتى بعد ذلك السؤال: هل هذه المشكلة لها حل تحليلى يمكن الوصول إليه دون الحاجة إلى خوارزمات التحليل العددى؟ إذا كانت المشكلة لها حل تحليلى فيحسن بل يجب اتباع هذا الحل تجنبا لمشاكل التقريب والدقة. أما إذا كان من الصعب الوصول إلى حل تحليلى فلا مجال إذن سوى الحل العددى.

سواء تم حل المشكلة بالطرق التحليلية أو بالطرق العددية فإنه يجب عمل محاكاة لهذا الحل على الحاسب، حيث تأتى بعد ذلك دراسة هذا الحل عند القيم المختلفة للمعاملات المستخدمة، ودراسة استجابة النظام للتغيير فى هذه المعاملات. تأتى بعد ذلك المرحلة الأخيرة من حل المشكلة وهى بناء النموذج الأولى لهذا المشروع ودراسة سلوكة فى الظروف المختلفة

للمعاملات وهل هى متوافقة مع ما تم الحصول عليه فى مرحلة المحاكاة أم لا. تأتى بعد ذلك المرحلة الأخيرة وهى مرحلة إنتاج النظام.



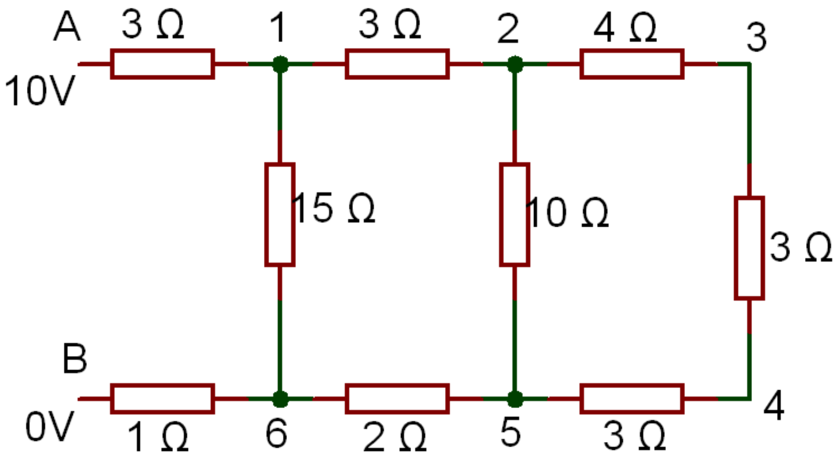
شكل ٣-١ خطوات حل أى مشكل أو تصميم أى نظام

لكى نرى أهمية ذلك أنظر مثلا إلى تصميم السيارة فإنه بوضع نموذج افتراضى لها على الحاسب يمكن دراسة أداء السيارة فى هذا الوسط الافتراضى والنظر فى كيفية استجابتها مثلا للرياح الشديدة من كل جانب لها، أو استجابتها عند عبور منحنيات الطريق ذات الدرجات المختلفة وتأثير نظام الكبح فى هذه الظروف، أو استجابة السيارة للمصادمات التى قد تطرأ عليها من اتجاهات أو مؤثرات مختلفة. نتيجة دراسة هذه المؤثرات المختلفة على أداء السيارة فى الوسط الافتراضى على الحاسب، يمكن إعادة تعديل بعض المعاملات فى نموذج السيارة الأصلى وإعادة الاختبارات فى هذا الوسط الافتراضى مرة أخرى، وهكذا يستمر الحال إلى أن يتم الوصول إلى الصورة النهائية للسيارة حيث يمكن بعدها إنتاج مرحلة النموذج الأولى

لها وإجراء الاختبارات المختلفة على هذا المنتج الأولى كمرحلة أخيرة للتاكد من صحة الحل أو التصميم. تخيل أنت مقدار التوفير فى الوقت والتكلفة بسبب إجراء كل هذه الاختبارات فى الوسط الافتراضى على الحاسب.
سنوضح ذلك أكثر بفرض المثال التالى على حل دائرة كهربية بسيطة حيث يمكن اتباع نفس الأسلوب مع أى دائرة أكثر تعقيدا من ذلك.

مثال ٣-١

إحسب التيار الكهربى فى كل فرع من أفرع الدائرة الموضحة فى شكل ٣-٢ وكذلك الجهد عند كل عقدة فى نفس الدائرة.



شكل ٣-٢ دائرة المثال ٣-١

هذه الدائرة تحتوى تسعة مقاومات قيمها موضحة فى الشكل ويتم تغذيتها من مصدر جهد مستمر قيمته ١٠ فولت. سنستخدم فى تحليل هذه الدائرة قانون أوم الذى ينص على أن التيار المار فى أى مقاومة يساوى فرق الجهد على هذه المقاومة مقسوما على قيمة هذه المقاومة. فمثلا التيار المار فى المقاومة ٣أوم التى بين العقدتين ١ و ٢ والمار من العقدة ١ فى اتجاه العقدة ٢ يساوى جهد النقطة ١ ناقص جهد النقطة ٢ مقسوما على قيمة هذه المقاومة (٣أوم). القانون الثانى الذى سنستخدمه هو قانون كيرتشوف للتيار والذى ينص على أن مجموع التيارات الداخلة والخارجة عند أى نقطة يساوى صفر. بتطبيق هذين القانونين على النقاط من ١ حتى ٦ فى الشكل ٣-٢ على التوالى مع بعض الاختصارات نحصل على المعادلات التالية:

$$11V_1-5V_2-V_6=50 \qquad (٣-١)$$

$$-20V_1+41V_2-15V_3-6V_5=0 \qquad (٣-٢)$$

$$-3V_2+7V_3-4V_4=0 \qquad (٣-٣)$$

$$-V_3+2V_4-V_5=0 \qquad (٣-٤)$$

$$-3V_2-10V_4+28V_5-15V_6=0 \qquad (٣-٥)$$

$$-2V_1-15V_5+47V_6=0 \qquad (٣-٦)$$

المعادلات من (٣- ١) حتى (٣- ٦) تعتبر النموذج الحسابى الخاص بهذه الدائرة. بالطبع من الممكن حل هذه المعادلات الستة تحليليا وإيجاد قيم للجهود عند النقاط الستة التى فى الدائرة وإيجاد التيارات فى كل الأفرع أيضا. ولكن ماذا لو كانت هذه الدائرة تحتوى على المئات وربما الآلاف من النقاط والأفرع. فى هذه الحالة سيكون نموذج الدائرة معقد جدا ومن الصعب حله تحليليا، وفى هذه الحالة لابد من البحث عن خواريزم عددى لحل مثل هذا النموذج، وهذا ما سنراه فى الفصول التالية. بعد الوصول إلى حل للنظام وحساب الجهود والتيارات المطلوبة، يمكن التحقق من سلوك الدائرة عند القيم المختلفة للمقاومات وجهد الدخل المستمر، وهذا يتم عمله بسهولة على النموذج وتنفيذ البرنامج على الحاسب والحصول النتائج مباشرة. فى النهاية وبعد دراسة الدائرة على النموذج الافتراضى يمكن بناء النموذج الأولى للدائرة من مقاومات حقيقية ومصدر جهد حقيقى ودراسة أداء الدائرة فى العالم الحقيقى.

الدائرة السابقة كانت مبسطة جدا، فماذا لو احتوت هذه الدائرة على مكثفات وملفات حيث فى هذه الحالة سندخل فى مجال المعادلات التفاضلية والتكاملية وسيصبح نموذج الدائرة أكثر تعقيدا، وأما إذا احتوت الدائرة على عناصر فعالة مثل الدايودات والترانزستورات فإنها ستكون أكثر تعقيدا، ومن هنا تظهر أهمية اللجوء إلى الحلول العددية.

٣-٣ أخطاء الحساب العددى

فى هذا المجال سنستخدم بعض المصطلحات التى لابد من تحديدها قبل الدخول فى أى تفاصيل. من هذا المصطلحات مايلى: الخطأ error، والدقة accuracy، والانضباط precession.

الخطأ

الخطأ هو مقدار الفرق بين الإجابة التى نحصل عليها كحل للمشكلة أو مقدار الكمية المقاسة فى أجهزة القياس أو أحيانا تسمى الكمية الفعلية والقيمة الحقيقية للمتغير المقاس. لذلك يمكن التعبير عن مقدار الخطأ بالمعادلة التالية:

$$\varepsilon = |X - Y| \qquad (٣-٧)$$

حيث X هى القيمة المقاسة أو الفعلية، و Y هى القيمة الحقيقية أو المتوقعة، و ε هى القيمة المطلقة للخطأ.

يعرف الخطأ النسبى بأنه القيمة المطلقة للخطأ منسوبة أو مقسومة على القيمة الحقيقية للمتغير المراد قياسه، وبالتالى يمكن التعبير عن النسبة المئوية للخطأ كالتالى:

$$\%\varepsilon = \frac{\varepsilon}{Y} 100\% \qquad (٣-٨)$$

بالتعويض من المعادلة (٣- ٧) فى المعادلة (٣- ٨) يمكن كتابة النسبة المئوية للخطأ كما يلى:

$$\%\varepsilon = \left|\frac{X-Y}{Y}\right| 100\% \qquad (٣-٩)$$

فمثلا بفرض أن درجة الحرارة الحقيقية هى ٣٠ درجة مئوية، ولكن عند قياسها بالترمومتر كانت القراءة ٣٥ درجة مئوية نتيجة وجود الترمومتر فى الشمس. فى هذه الحالة يكون الخطأ يساوى مقدار الفرق بين القيمة الفعلية المقاسة للحرارة (٣٥ درجة) وقيمتها الحقيقية (٣٠ درجة) ولذلك يمكن كتابة مقدار الخطأ كما يلى:

مقدار الخطأ= | ٣٠-٣٥ | = ٥ درجات

وتكون النسبة المئوية للخطأ هى:

$$\%\varepsilon = \left|\frac{30-35}{30}\right| 100\% = \frac{100}{6} = 16.666\ldots\% \qquad (٣-١٠)$$

الدقة

تعرف الدقة accuracy بأنها مقدار قرب الكمية الفعلية أو المقاسة للمتغير من قيمته الحقيقية. لذلك فإن الدقة النسبية يمكن التعبير عنها كما يلى:

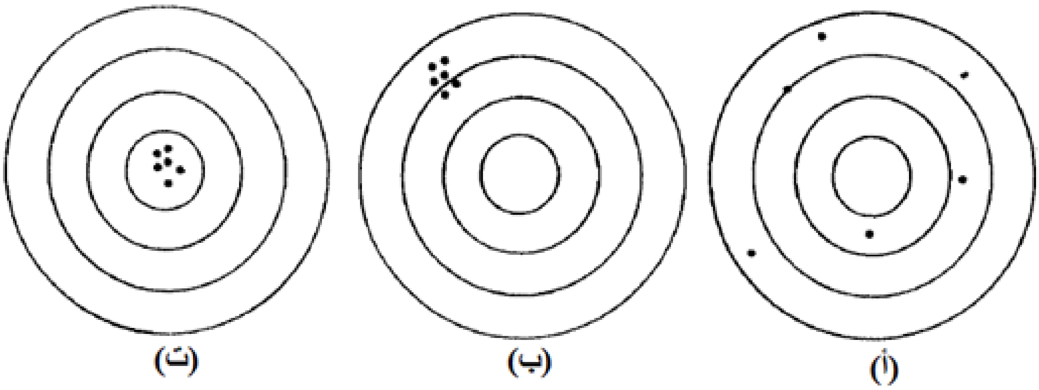
$$A = 1 - \left|\frac{X-Y}{Y}\right| \qquad (٣-١١)$$

حيث A هى الدقة النسبية. بالتالى يمكن كتابة النسبة المئوية للدقة كالتالى :

$$\%A \equiv 100 - \%\varepsilon = A \times 100$$

بالنسبة لمثال درجة الحرارة السابق ستكون دقة القياس تساوى=١٠٠-١٦,٦٦٦=٨٣,٣٣٤%.

الانضباط



شكل ٣-٣ إجراء ستة قياسات تحت نفس الظروف لثلاثة أجهزة قياس مختلفة، (أ) جهاز قليل الدقة، قليل الانضباط، (ب) جهاز قليل الدقة عالى الانضباط، (ت) جهاز عالى الدقة عالى الانضباط .

الانضباط precision هو مقياس لمقدرة جهاز القياس أن يعطى نفس القيمة المقاسة مع تكرار عملية القياس تحت نفس الظروف. شكل ٣-٣ يبين ناتج أو خرج ثلاث أجهزة قياس مختلفة. القيمة الحقيقية أو المتوقعة من كل جهاز قياس هى مركز الدوائر الموضحة فى كل شكل ، والنقط السوداء تمثل النتيجة أو القيمة الفعلية مع تكرار عملية القياس. شكل ٣-٣أ يمثل خرج الجهاز الأول. فى هذا الشكل تم أخذ ستة قياسات تحت نفس الظروف وبنفس الجهاز الأول. نلاحظ من هذا الشكل أن كل القياسات متباعدة عن بعضها لذلك لا يوجد انضباط فى هذا الجهاز. نلاحظ أيضا أن القراءات الناتجة من هذا الجهاز بعيدة عن القيمة الحقيقية التى هى مركز الدوائر، لذلك فإن دقة هذا الجهاز قليلة جدا. لذلك نستطيع القول بأن هذا الجهاز الأول قليل الانضباط، قليل الدقة low precision, low accuracy. شكل ٣-٣ب يمثل خرج

الجهاز الثانى. فى هذا الشكل تم أخذ ستة قياسات أيضا تحت نفس الظروف وبنفس الجهاز الثانى. نلاحظ من هذا الشكل أن كل القياسات قريبة جدا من بعضها، لذلك نستطيع القول بأن هذا الجهاز منضبط جدا . نلاحظ أيضا أن القراءات الناتجة من هذا الجهاز بعيدة عن القيمة الحقيقية التى هى مركز الدوائر، لذلك فإن دقة هذا الجهاز قليلة جدا . لذلك نستطيع القول بأن هذا الجهاز الثانى عالى الانضباط، قليل الدقة high precision, low accuracy. غالبا هذا الجهاز يعانى من مشكلة انحياز يجعل كل قرآته بعيدة عن القيمة الحقيقية، لذلك إذا تم إصلاح هذا الانحياز (أو إعادة معايرته) الذى قد يكون نتيجة ارتفاع فى درجة الحرارة فإن الجهاز سيصبح عالى الدقة عالى الانضباط. نشبه ذلك بفرض أنك وزميلك متفقان على التقابل كل يوم الساعة السابعة صباحا، ولكن المشكلة أن زميلك يأتى كل يوم الساعة ٧,١٠ أى أنه يأتى دائما متأخرا عشرة دقائق. هذا الزميل نستطيع القول عنه أنه غير دقيق حيث أنه يأتى متأخرا عن موعده بمقدار ١٠ دقائق كاملة، ولكنه منضبط لأنه يأتى كل يوم فى نفس الموعد. لو نظرنا فى أحوال هذا الزميل ربما نجد أن المنبه الخاص به متأخرا بمقدار عشرة دقائق، وهذا ما يسمى انحياز فى المنبه. شكل ٣-٣ت يمثل خرج الجهاز الثالث. فى هذا الشكل تم أخذ ستة قياسات أيضا تحت نفس الظروف وبنفس الجهاز الثالث. نلاحظ من هذا الشكل أن كل القياسات قريبة جدا من بعضها، لذلك نستطيع القول بأن هذا الجهاز منضبط جدا. نلاحظ أيضا أن القراءات الناتجة من هذا الجهاز قريبة جدا من القيمة الحقيقية التى هى مركز الدوائر، لذلك فإن هذا الجهاز عالى الدقة. لذلك نستطيع القول بأن هذا الجهاز الثالث عالى الانضباط، عالى الدقة high precision, high accuracy.

بعد تعرفنا على الخطأ والدقة والانضباط والفرق بينها سنبدأ فى التعرف على الأخطاء الموجودة فى الحاسب وسبب هذه الأخطاء. ربما يقول البعض لماذا القلق على أخطاء الحاسب التى تكون عادة صغيرة جدا يمكن إهمالها. فى الحقيقة أن القلق ليس على القيمة المطلقة للخطأ ولكن المشكلة هى تواجد هذه الأخطاء فى خواريزمات تتكرر آلاف وربما ملايين من المرات حيث فى هذه الحالة يحدث تراكم لهذا الخطأ مما يكون له تأثير لا يمكن إهماله وربما يؤدى إلى كوارث.

٣-٤ أخطاء الحساب الرقمى

سنعرض هنا لبعض مصدار الخطأ فى الحساب الرقمى التى يمكن أن تظهر نتيجة استخدام الحاسب فى الحل الرقمى لأى مشكلة.

خطأ التقريب

يحدث هذا الخطأ نتيجة محدودية عدد الخانات المستخدمة فى التعبير عن أى رقم. فمثلا الماتلاب يعبر عن الأرقام باستخدام ١٦ خانة فقط والخانات التى تكون خارج نطاق هذا العدد من الخانات ستمثل الخطأ فى التعبير عن هذا الرقم. الأمر format فى ماتلاب يحدد كيفية عرض النتيجة على الشاشة، وهناك نوعان من هذا الأمر، الأول format short يعرض النتيجة فى خمسة خانات عشرية، والنوع الثانى format long يعرض النتيجة فى ١٦ خانة عشرية وهذا أقصى عدد من الخانات فى الماتلاب. أنظر كيف يعرض ماتلاب النسبة π فى كل من الشكلين:

```
>> format long
>> pi
```

```
ans =
  3.141592653589793
```

لاحظ أنه بمجرد كتابة pi (وهى كيفية التعبير عن النسبة π فى ماتلاب) فإن ماتلاب يرد بكتابة قيمة π فى ١٦ خانة عشرية من الدقة ويضعها فى متغير تلقائى من عنده وهو المتغير ans. الآن دعنا نغير عرض النتيجة إلى خمسة خانات عشرية بدلا من ١٦ كما يلى:

```
>> format short
>> pi
ans =
   3.1416
```

لاحظ أن تخفيض عدد الخانات العشرية من ١٦ إلى خمسة خانات من الممكن أن يتم بطريقتين، الأولى هى القطع chop حيث يتم القطع من بعد الخانة الخامسة مباشرة، وفى هذه الحالة كانت النتيجة ستكون ans=3.1415. الطريقة الثانية هى تقريب الخانة السادسة لأقرب قيمة. الخانة السادسة تساوى 9 فى هذه الحالة، أى أكبر من 5، لذلك يتم زيادة واحد على الخانة الخامسة فأصبحت 6 بدلا من 5 لذلك حصلنا على القيمة ans=3.1416 بدلا من ans=3.1415. عندما تكون الخانة السادسة أقل من خمسة لا يتم زيادة واحد على الخانة الخامسة وتبقى كما هى.

لاحظ أنه عند عرض النتيجة فى خمس خانات فإن ذلك لا يعنى أن الماتلاب أثناء إجراء الحسابات يستخدم هذه الخانات الخمسة، ولكنه يستخدم تمثيل الأرقام من ١٦ خانة فى كل الحسابات الداخلية التى تجرى بداخله، فقط عند عرض النتيجة يتم عرضها إما فى التشكيل الطويل من ١٦ خانة أو التشكيل القصير من خمسة خانات. هناك أوامر تشكيل أخرى لعرض النتائج مثل format longe و format shorte فيرجى الرجوع إلى ماتلاب للوقوف على حقيقتها.

أنظر إلى الأمر التالى:

```
>> format short
>> sqrt(3)
ans =
   1.7321
```

بعد الأمر format short لنعرض النتيجة فى خمس خانات عشرية طلبنا من ماتلاب أن يعرض لنا الجذر التربيعى للرقم 3 فكانت النتيجة هى 1.7321. الآن سنأخذ هذه النتيجة ونطلب من ماتلاب أن يربعها مرة أخرى حيث من المفروض أن نحصل على الرقم 3، ولكننا لم نحصل على 3 بالضبط:

```
>> (1.7321)^2
ans =
   3.0002
```

لقد حصلنا على 3.0002 بدلا من 3 لأننا قمنا بتربيع الرقم المقرب.

الآن سنطلب من ماتلاب الجذر التربيعى للرقم 3 مرة أخرى ثم نقوم بتربيع النتيجة التى حصلنا منه مباشرة مرة أخرى كما يلى حيث سنرى النتيجة هى 3.0000 مقربة لأقرب خمسة خانات:

```
>> sqrt(3)
```

```
ans =
   1.7321
>> ans^2
ans =
   3.0000
```

بهذا نتأكد أن ماتلاب يستخدم فى حساباته الخانات المطولة من ١٦ خانة.

سنعيد الأوامر السابقة ولكن بعد الانتقال إلى الشكل الطويل من الخانات ١٦ خانة كما يلى:

```
>> format long
>> sqrt(3)
ans =
  1.732050807568877
>> ans^2
ans =
  3.000000000000000
>> (1.732050807568877)^2
ans =
  2.999999999999999
```

شكل ٣-٥ سقوط الصاروخ آريان 5 بعد إطلاقه مباشرة نتيجة أخطاء فى التقريب الحسابى

شكل ٣-٤ الصاروخ باترويت الذى سقط فى موقعه نتيجة خطأ التقريب الحسابى

خطأ القطع

القطع هنا لا يكون فى عدد الخانات التى يتم تمثيل الأرقام فيها ولكن القطع يكون فى عدد الكميات الممثلة لمتغير معين كما فى حالة استخدام المفكوك أو المتواليات لتمثيل دوال معينة، مثل مفكوك تايلور Taylor series لتمثيل الدالة e^x كما يلى:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} \dots \qquad (١٢-٣)$$

حيث تم هنا التعبير عن الدالة e^x بعدد لا نهائى من الكميات، وعند استخدام هذا التعبير فى بعض الخوارزمات يتم القطع عند عدد معين من الكميات كأن يتم الاكتفاء بأول ثلاث أو أربع أو خمسة كميات توفيرا لوقت الحساب. بالطبع ستتوقف دقة التعبير وبالتالى الخطأ على عدد الكميات التى سيتم القطع عندها.

أخطأ التقريب الحسابى

عند نمذجة أو محاكاة أى مشكلة أو أى تصميم نلجأ إلى استخدام النمذجة الخطية بديلا عن النمذجة غير الخطية لتبسيط نموذج محاكاة المشكلة أو وقت الحساب لها. ينتج التقريب الحسابى أيضا من استخدام عدد محدد من البتات لتمثيل الأرقام. بالطبع فإن ذلك سيتسبب فى أخطاء من الممكن أن تكون كبيرة وكارثية.

كمثال على الكوارث التى تم إرجاعها إلى أخطاء فى التقريب الحسابى هو حادثة الصاروخ باترويت الأمريكى الشهيرة فى قاعدة الظهران ٢٥ فبراير من عام ١٩٩١ حيث بعد إطلاق الصاروخ سقط فى نفس القاعدة مما نتج عنه ٢٨ قتيلا. شكل ٣-٤ يبين رسما لهذا الصاروخ. ولقد تم إرجاع سبب ذلك إلى أخطاء فى قياس الزمن نتيجة النزول فى تمثيل الأرقام فى نظام النقطة العائمة الذى سنراه بعد قليل من ٣٢ بت إلى ٢٤ بت، مما نتج عنه خطأ تقريب حسابى من الممكن أن يتزايد داخل الخوارزم المستخدم. أنظر الموقع لتفاصيل أكثر عن أخطاء الحاسب(informatik. http://wwwzenger. /bugse. huckle tu-muenchen.de/persons/ html).

المثال الثانى هو مثال الصاروخ آريان 5 الذى انفجر بعد أطلاقه مباشرة فى ٤ يناير من عام ١٩٩٦ فى كارثة كانت الأولى من نوعها فى ذلك الوقت لهذا النوع من الصواريخ وقد تم إرجاع ذلك أيضا إلى تمثيل الأرقام فى ١٦ بت بدلا من ٦٤ بت مما نتج عنه خطأ فى التقريب الحسابى أدى إلى هذه الكارثة. شكل ٣-٥ يبين شكلا لهذا الصاروخ لحظة إطلاقه.

خطأ الإلغاء

نظام النقطة العائمة floating point لتمثيل الأرقام

أى رقم حقيقى يمكن تمثيلة بتتابع لا نهائى من الأرقام العشرية، ولنضرب لذلك المثال التالى لرقم كسرى غير منتهى كما يلى:

$$\frac{8}{3} = 2.666\ldots = \left(\frac{2}{10^1} + \frac{6}{10^2} + \frac{6}{10^3} + \frac{6}{10^4} + \cdots\right) X10^1 \qquad (١٣-٣)$$

المعادلة (٣- ١٣) تمثل تتابع لا نهائى من الخانات غير المنتهية، وبالطبع فإن الحاسب لن يستطيع تمثيل عدد لا نهائى من الخانات لذلك لابد من استخدام القطع أو التقريب بعد عدد معين من الخانات، حيث يمكن القطع بعد ثلاث خانا عشرية مثلا، فيصبح تمثيل الرقم $\frac{8}{3}$ بعد ثلاث خانات عشرية كما يلى:

$$\frac{8}{3} \cong 2.666 \cong \left(\frac{2}{10^1} + \frac{6}{10^2} + \frac{6}{10^3} + \frac{6}{10^4}\right) X10^1 \qquad (١٤-٣)$$

بذلك يمكن وضع صورة عامة لتمثيل أى رقم فى نظام النقطة العائمة لتمثيل الأرقام كما يلى:

$$x \cong \pm 0.d_1 d_2 d_3 \ldots d_{t-1} d_t X10^e \qquad (١٥-٣)$$

$$x \cong \pm\left(\frac{d_1}{10^1} + \frac{d_2}{10^2} + \frac{d_3}{10^3} \ldots + \frac{d_{t-1}}{10^{t-1}} + \frac{d_t}{10^t}\right) X10^e \qquad (١٦-٣)$$

حيث يتم اختيار الأس e والدقة t التى سيتم القطع عندها بحيث يكون الرقم x أقرب ما يكون لقيمته الحقيقية.
فى تمثيل الرقم $\frac{8}{3}$ السابق كان الأس e يساوى ١ و t تساوى ٣ حيث تم تمثيل الرقم لأقرب ثلاث خانات عشرية. التمثيل السابق للرقم $\frac{8}{3}$ ليس فريدا حيث يمكن كتابته بأكثر من طريقة كما يلى:

$$\frac{8}{3} = 2.666X10^0 = 0.2666X10^1 = 0.02666X10^2 = 0.002666X10^3$$

لذلك كان لابد من توحيد طريقة الكتابة وبالذات عند التعامل داخل الحاسبات لذلك تم الاتفاق على المعيار القياسى IEEE 754 الذى تستخدمه جميع الحاسبات الرقمية والذى تم فيه كتابة الأرقام الحقيقية فى نظام النقطة العائمة فى ثلاث مجالات هى الإشارة، ومقدار الرقم الذى لابد أن تكون فيه الخانة التى على يسار العلامة العشرية تساوى 0 وأول خانة على يمين العلامة لا تساوى 0، ثم بعد ذلك يأتى المجال الثالث وهو الأس الذى يمثل القوة التى يرفع إليها الرقم 10 الذى هو قاعدة النظام العشرى. لذلك فإن الصورة القياسية للرقم $\frac{8}{3}$ ستكون $0.2666X10^1$. وفى هذه الحالة نقول أن الرقم قد تم تقريبه لأربع خانات عشرية ولا نقول أنه تم تقريبه لثلاث خانات عشرية على يمين العلامة كما سبق. نظام النقطة العائمة يتم تمثيله فى النظام الثنائى بطريقتين تمثلان دقتين مختلفتين كما يحدث فى برنامج الماتلاب:

الطريقة الأولى (تسمى طريقة الدقة الأحادية single precession) تستخدم ٣٢ بت مقسمة على الثلاث مجالات كما يلى:

- بت واحدة تمثل الإشارة تساوى 1 إذا كان الرقم سالب، و 0 إذا كان الرقم موجب وهى البت التى فى أقصى يسار الرقم.
- يأتى بعد ذلك ثمانية بتات تمثل الأس وهو القوة التى سيرفع لها الرقم 2 حيث 2 هى قاعدة النظام الثنائى.
- ثم بعد ذلك هناك ٢٣ بت تمثل مقدار الرقم على أن تكون أول بت فيه لا تساوى 0.

الطريقة الثانية (تسمى طريقة الدقة المضاعفة double precession) تستخدم 64 بت مقسمة على الثلاث مجالات كما يلى:

- بت واحدة تمثل الإشارة تساوى 1 إذا كان الرقم سالب، و 0 إذا كان الرقم موجب وهى البت التى فى أقصى يسار الرقم.
- يأتى بعد ذلك ١١ بت تمثل الأس وهو القوة التى سيرفع لها الرقم 2 حيث 2 هى قاعدة النظام الثنائى.
- ثم بعد ذلك هناك ٥٢ بت تمثل مقدار الرقم على أن تكون أول بت فيه لا تساوى 0.

عند التحويل من النظام الثنائى للنظام العشرى فإننا نستخدم كل ٤ خانات ثنائية لتمثل خانة واحدة عشرية لذلك فإن النظام أحادى الدقة المكون من ٣٢ خانة يتحول إلى ثمانية خانات عشرية فى النظام العشرى وأما النظام المضاعف الدقة المكون من ٦٤ بت فيتحول إلى ١٦ خانة عشرية. التعامل مع الأرقام ذات النقطة العائمة من الممكن أن ينشأ عنه نوع آخر من الأخطاء بالذات عند طرح كميتين تقترب كل منهما من الأخرى بدرجة كبيرة. أنظر للبرنامج التالى بلغة ماتلاب:

```
>> x=pi;
>> y=6.022e23;
>> z=x+y-y
z =
```

0

فى هذا البرنامج البسيط تم وضع $x=\pi=3.14...$، ووضع y تساوى رقم أفوجادرو الممثل فى نظام النقطة العائمة بالقيمة $6.022x10^{23}$ وهو رقم كبير جدا كما نرى. بجمع الرقمين x و y وهما الرقم y الكبير جدا كما نرى والرقم x الصغير جدا كما نرى أيضا بالنسبة للرقم y، فإن النتيجة ستكون هى y تقريبا لأن إضافة x ستكون بمثابة خطأ بالنسبة للرقم y خارج حدود دقة الماتلاب الممثلة بعدد ١٦ خانة عشرية. لذلك فإن x+y ستساوى y تقريبا. عند تقدير قيمة المتغير z فى السطر الثالث من البرنامج وحسب أولويات الحساب فى ماتلاب كلغة برمجة فإنه سيجمع x مع y أولا ثم يطرح من هذا المجموع القيمة y ولذلك كانت النتيجة تساوى صفر. إن ذلك بالطبع بعيد عن القيمة الصحيحة حيث من المفروض نظريا أن تكون النتيجة هى $z=\pi$.

لقد نشأ الخطأ السابق نتيجة طرح $y=6.022x10^{23}$ من الكمية $(x+y)=(3.14...+6.022x10^{23})$ وكليهما تمثل كمية كبيرة جدا ومتاقربتان جدا، لذلك كانت نتيجة الطرح صغيرة جدا بحيث لا يمكن التعبير عنها فى ماتلاب بالدقة المتاحة، ولذلك كانت النتيجة خطأ وتساوى صفر. مثل هذا الخطأ يسمى خطأ الإلغاء cancellation error. المثال التالى سيوضح هذا الخطأ بطريقة أكثر وضوحا.

مثال ٣-٢

لماذا تكون كتابة الكمية (x^2-y^2) على الصورة $(x+y)(x-y)$ طريقة مفضلة للتخلص من خطأ الإلغاء؟

سنفترض القيم التالية لكل من x و y لكى نسهل الأمور. إفترض أن $x=9000.2$ وأن $y=9000.1$. بالتعويض عن x و y فى الكمية (x^2-y^2) سنحصل على القيمة الحقيقية لهذا الفرق وهى:

$$(x^2-y^2)=[(9000.2)^2-(9000.1)^2)=1800.03.$$

الآن سنستخدم التعبير القياسى IEEE 754 للنقطة العائمة مع استخدام دقة مقدارها ٥ خانات كما يلى: $x=9.0002x10^3$ و $y=9.0001x10^3$ و $x^2=81003600.04$ وعند التعبير عنها بنظام النقطة العائمة ذات الخمس خانات ستصبح $x^2=8.1003x10^7$. بنفس الطريقة $y^2=81001800.01$ ، وعند التعبير عنها بنظام النقطة العائمة ذات الخمس خانات ستصبح $y^2=8.1001x10^7$. وعلى ذلك يمكن كتابة الفرق كما يلى:

$$(x^2-y^2)= 8.1003x10^7-8.1001x10^7=0.0002x10^7$$

وعند التعبير عن هذه النتيجة بنظام النقطة العائمة القياسى ذو الخمس خانات بحيث يكون الرقم الذى على يسار العلامة لا يساوى الصفر ستصبح:

$$(x^2-y^2)=2.0000x10^3$$

أى أن نتيجة الفرق (x^2-y^2) فى هذه الحالة أصبحت تساوى 2000 وهذا بعيدا جدا عن القيمة الحقيقية أو النظرية التى حصلنا عليها مسبقا وهى 1800.03. أى أن الخطأ النسبى سيكون فى هذه الحالة يساوى:

الخطأ النسبى = $|2000-1800.03|/1800.03 = 11.1\%$

وهذه بالطبع نسبة خطأ كبيرة جدا. هذا الخطأ (خطأ الإلغاء) نشأ لسببين، أولا بسبب التعامل مع نظام النقطة العائمة فى التعبير عن الأرقام، وثانيا نتيجة التعامل مع عدد محدود من الخانات.

الآن دعنا نستخدم الصورة الأخرى للكمية (x^2-y^2) وهى $(x+y)(x-y)$، وبالتعويض عن x و y فى كل منهما باستخدام نظام النقطة العائمة ذات الخمس خانات نحصل على ما يلى:

$$x+y=9.0002x10^3+9.0001x10^3= 1.8000x10^3$$

$$x-y=9.0002x10^3-9.0001x10^3= 0.0001x10^3$$

$$(x+y)(x-y)=1.8x10^3x0.0001x10^3 =0.001800x10^6= 1.8x10^3 =1800$$

بمقارنة هذه النتيجة (1800) مع النتيجة الحقيقية أو النظرية (1800.03) نجد أن الخطأ الناتج أصبح صغيرا جدا هذه المرة ويمكن كتابة الخطأ النسبى فى هذه المرة كما يلى:

الخطأ النسبى = $|1800-1800.03|/1800.03 = 0.00167\%$

لقد أكد هذا المثال على حقيقة أن خطأ الإلغاء ينشأ فى الأساس من حساب الفرق بين كميتين كبيرتين جدا الفرق بينهما صغيرا جدا. ولقد أكد هذا المثال أيضا على أن هذا الخطأ يمكن تقليله بقدر كبير عن طريق إعادة صياغة الكمية المراد حسابها بما لا يؤثر على قيمتها الحقيقية حيث كما رأينا أن إعادة صياغة الكمية (x^2-y^2) على الصورة $(x+y)(x-y)$ قد قلل من نسبة الخطأ الناتج بدرجة كبيرة جدا. أنظر أيضا إلى المثال التالى.

مثال ٣-٣

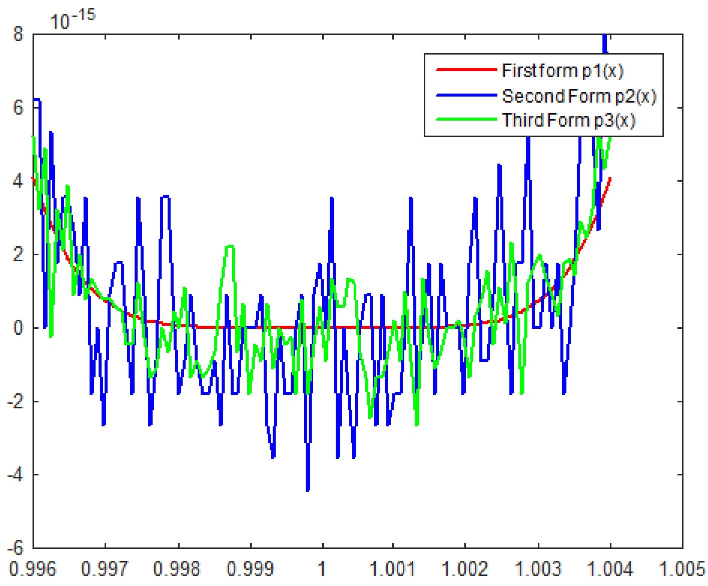
كث يرة الحدود $p(x)=(x-1)^6$ من الممكن أن تؤدى إلى خطأ الإلغاء عند حسابها بالتعويض عن قيمة x بقيمة قريبة جدا من الواحد. يمكن كتابة كثيرة الحدود تلك بثلاث طرق مختلفة كما يلى:

$$p(x)=p_1(x)=(x-1)^6 \quad (٣-١٧)$$

$$p(x)=p_2(x)=x^6-6x^5+15x^4-20x^3+15x^2-6x+1 \quad (٣-١٨)$$

$$p(x)=p_3(x)=1+x(-6+x(15+x(-20+x(15+x(-6+x))))) \quad (٣-١٩)$$

فأى الصور الثلاثة لكثيرة الحدود p(x) ستكون الأدق أو الأقل فى مقدار خطأ الإلغاء. لكى نرى هذا الفرق دعنا نرسم قيمة كل كثيرة حدود فى الصور الثلاثة المتكافئة السابقة مع تغيير x من 0.996 حتى 1.004 حيث سنرسم ١٠٠ نقطة فى هذا المدى مستخدمين فى ذلك البرنامج التالى بلغة الماتلاب.



شكل ٣-٦ تأثير الصور المختلفة لكثيرة الحدود p(x) على خطأ الإلغاء

```
% demonstration on numeric cancellation problem
x=linspace(0.996,1.004);%take 100 value for x between 0.996 and 1.004
p1=(x-1).^6;
p2=x.^6-6*x.^5+15*x.^4-20*x.^3+15*x.^2-6*x+1;
p3=1+x.*(-6+x.*(15+x.*(-20+x.*(15+x.*(-6+x)))));
plot(x,p1,'r',x,p2,'b',x,p3,'g', 'LineWidth',1.5);
legend('First form p1(x)', 'Second Form p2(x)', 'Third Form p3(x)')
```

كما نلاحظ من شكل ٣-٦ أن الصورة الأولى $p_1(x)$ التى تظهر باللون الأحمر فى الشكل هى الأكثر دقة والأقل فى خطأ الإلغاء حيث أن الصورتين $p_2(x)$ و $p_3(x)$ باللونين الأزرق والأخضر على التوالى يعطيان خطأ أكثر وعدم استقرار أو تأرجح كبير فى القيمة كما نرى.

سنقدم شرحا سريعا لكل أمر من أوامر الماتلاب التى ترد لأول مرة وسنترك للقارىء الاستزادة عن الماتلاب بالرجوع إلى أى مرجع فيه أو كتاب "تعلم ماتلاب بنفسك" باللغة العربية من مركز الترجمة لجامعة الملك سعود ٢٠١٣.

فى الماتلاب أى نص مكتوب مسبوق بالعلامة % حتى نهاية السطر يعتبر تعليقا وليس أمرا للتنفيذ، وفى العادة يكتبه محرر ماتلاب باللون الأخضر كما فى السطر الأول من البرنامج السابق ونهاية السطر الثانى. الأمر x=linspace(0.996, 1.004) سيقوم بتقسيم المسافة بين 0.996 (البداية) و 1.004 (النهاية) مستخدما ١٠٠ نقطة على أبعاد متساوية. الصورة العامة لهذا الأمر هى x=linspace(x1, x2, N) حيث x1 هى نقطة البداية و x2 هى نقطة النهاية و N هى عدد نقاط التقسيم التى تساوى ١٠٠ كوضع تلقائى فى عدم وجود N. أكتب help linspace ستجد أن ماتلاب يكتب لك رسالة مساعدة كافية جدا عن هذا الأمر. بعد ذلك تم كتابة الصور الثلاثة لكثيرة الحدود p1(x) و p2(x) و p3(x). لاحظ استخدام عملية النقطة "." فى المعادلات الثلاثة، حيث أن المتغير x عبارة عن متجه من ١٠٠ قيمة نتيجة الأمر linspace، فإنه

لكى يتم حساب الأس لكل قيمة من قيم المتجه x لابد من استخدام عملية النقطة. مثلا (x-1).^6 سيحسب القوس (x-1) ثم يرفعه للأس 6 عند كل قيمة من قيم المتغير x. وأيضا x.^5 سيحسب x مرفوعة للأس 5 عند كل قيمة من قيم المتجه x، وهكذا. يأتى بعد ذلك أمر الرسم التالى:

```
plot(x,p1,'r',x,p2,'b',x,p3,'g', 'LineWidth',1.5);
```

الذى سيرسم المنحنى x مع p1 باللون الأحمر 'r'، والمنحنى x مع p2 باللون الأزرق 'b'، و المنحنى x مع p3 باللون الأخضر 'g'، وسيكون عرض (سمك) الخط المستخدم 'LineWidth' يساوى 1,5. هناك معاملات وتفاصيل أخرى عن الأمر plot يمكنك الرجوع إليها بطلب المساعدة help plot. الأمر الأخير فى هذا البرنامج هو الأمر:

```
legend('First form p1(x)', 'Second Form p2(x)', 'Third Form p3(x)')
```

وهذا الأمر سيعطى مفتاح الشكل legend المكتوب فى مربع أعلى يمين الشكل. حيث هناك ثلاث منحنيات فإن ماتلاب سيعطيك فرصة لكتابة إسم لكل منحنى مثل 'First Form p1(x)' سيضعه ماتلاب بجوار خط بنفس لون المنحنى كما فى الشكل، والمنحنى الثانى أسميناه 'Second Form p2(x)'، والمنحنى الثالث أسميناه 'Third Form p3(x)'.

مثال ٣-٤

إفترض المعادلة التالية:

$$y = \sqrt{x+\delta} - \sqrt{x} \tag{٣-٢٠}$$

بفرض أن δ كمية صغيرة جدا موجبة، فإن الكميتين $\sqrt{x}$ و $\sqrt{x+\delta}$ سينشأ عنهما خطأ إلغاء بالذات عندما تكون δ صغيرة جدا حيث سيكون الفرق بين الجذرين صغيرا جدا. للتخلص من هذا الخطأ يمكن إعادة كتابة المعادلة السابقة بضربها فى كمية تمثل المرافق لها كما يلى:

$$y = \frac{\sqrt{x+\delta}+\sqrt{x}}{\sqrt{x+\delta}+\sqrt{x}}\left(\sqrt{x+\delta} - \sqrt{x}\right) \tag{٣-٢١}$$

بإجراء بعض الاختصارات نحصل على:

$$y = \frac{\delta}{\sqrt{x+\delta}+\sqrt{x}} \tag{٣-٢٢}$$

وهذه الصورة الأخيرة لنفس المعادلة y لن تكون عرضه لخطأ الإلغاء كما فى المعادلة (٣- ٢٠).

٣-٥ دقة الماكينة

أى ماكينة للحساب تقوم بالتعبير عن الأرقام الكبيرة وبالذات عند التعامل مع التطبيقات العلمية فى الصورة القياسية IEEE 754 فى عدد معين من الخانات. نتيجة التعبير فى هذا العدد من الخانات فإنه من الممكن أن يحدث خطأ تقريب كما رأينا فى الأمثلة السابقة. لذلك فإن دقة الماكينة أو دقة الحاسب machine precession تعتبر هى أصغر قيمة بين رقمين يمكن للحاسب أن يتعرف عليها ولا يعتبرها صفر. أو بمعنى آخر هى أصغر قيمة ε بحيث أن الفرق بين القيمة 1 والقيمة 1+ε لا تساوى الصفر، أو يمكن للحاسب أن يتعرف عليها. فى الحاسبات أحادية الدقة التى تعبر عن أرقام النقطة العائمة فى ٣٢ بت تكون دقة الماكينة تساوى 2^{-23} حيث يكون مقدار العدد معبر عنه فى ٢٣ بت كما رأينا، وهذه الدقة

تكون 10^{-7} فى النظام العشرى أى ما يعادل سبعة خانات عشرية. فى النظام المضاعف الدقة يتم التعبير عن الأرقام ذات النقطة العائمة فى ٦٤ بت منها ٥٢ بت للمقدار، ولذلك فإن دقة الماكينة تساوى 2^{-52} أى ما يعادل 10^{-16} أو ١٦ خانة عشرية. هذه الدقة ε أو إبسلون eps يعرفها ماتلاب ويمكن أن نطلب قيمتها منه كما يلى:

```
>> eps
ans =
2.220446049250313e-16
```

وهذه كما ذكرنا ستمثل أصغر فرق بين قيمتين يمكن للماتلاب أن يتعرف عليه ولا يعتبره صفرا. أنظر للأوامر التالية:

```
>> 1+eps
ans =
  1.000000000000000
>> 1+eps/2
ans =
   1
```

فى الأمر الأول طلبنا من ماتلاب أن يكتب 1+eps حيث eps هى دقة الماكينة أو هى أصغر فرق بين رقمين يعترف به الماتلاب، لذلك رد علينا الماتلا ب بالإجابة ans=1.0000000000000000 مما يعنى أنه اعترف بهذا الفرق ولكنه أصغر من أن يظهر فى الخانة السادسة عشرة لذلك رأينا ١٦ صفر بجوار الواحد. أما الأمر الثانى 1+eps/2 فإنه يجمع نصف الإبسلون مع الواحد، وحيث أن الماتلاب لا يعترف بالقيمة eps/2 فإنه يعتبرها صفر، ولذلك فقد أعطانا الإجابة ans=1 فى هذه الحالة. أى أن 1+eps>1 بينما 1+eps/2=1.

٣-٦ تمارين

١- الدالة $f_1(x.\delta)=\cos(x+\delta)-\cos(x)$ من الممكن أن تكون عرضه لخطأ الإلغاء عندما تكون δ صغيرة جدا. ولكن باستخدام بعض قواعد حساب المثلثات وبالذات العلاقة:

$$\cos(A)-\cos(B)=-2\sin\left(\frac{A+B}{2}\right)\sin\left(\frac{A-B}{2}\right)$$

(أ) أكتب صورة أخرى للعلاقة $f_1(x,\delta)$ بحيث تساويها تماما فى الحساب النظرى ولكنها تتجنب إمكانية حدوث خطأ الإلغاء فى الدالة $f_1(x,\delta)$. هذه العلاقة الجديدة سمها $f_2(x,\delta)$.

(ب) أكتب برنامج ماتلاب يحسب العلاقة: $g_1(x.\delta)=\frac{f_1(x.\delta)}{\delta}+sin(x)$ والعلاقة: $g_2(x.\delta)=\frac{f_2(x.\delta)}{\delta}+sin(x)$ عندما x=3 و $\delta=1\text{x}10^{-11}$.

(ت) إشرح الفرق فى قيمة كل من $g_1(x.\delta)$ و $g_2(x.\delta)$ فى الجزء (ب).

٢- الدالة $f_1(x.h)=\sin(x+h)-\sin(x)$ من الممكن أن تكون عرضه لخطأ الإلغاء عندما تكون h صغيرة جدا. ولكن باستخدام بعض قواعد حساب المثلثات وبالذات العلاقة:

$$\sin(A)-\sin(B)=2\cos\left(\frac{A+B}{2}\right)\sin\left(\frac{A-B}{2}\right)$$

(أ) أكتب صورة أخرى للعلاقة $f_1(x,h)$ بحيث تساويها تماما فى الحساب النظرى ولكنها تتجنب إمكانية حدوث خطأ الإلغاء فى الدالة $f_1(x,h)$. هذه العلاقة الجديدة سمها $f_2(x,h)$.

(ث) أكتب برنامج ماتلاب يحسب العلاقة: $g_1(x.h)=\frac{f_1(x.h)}{h}+cos(x)$ والعلاقة: $g_2(x.h)=\frac{f_2(x.h)}{h}+cos(x)$ عندما x=0 و $h=1x10^{-11}$.

(ج) إشرح الفرق فى قيمة كل من $g_1(x.h)$ و $g_2(x.h)$ فى الجزء (ب).

٣- العلاقة التالية: $S_n=\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$ تستخدم لتقريب المضروب n!=1x2x3….xn، للقيم الكبيرة للمتغير n. القيمة e فى التعبير S_n هى $e=e^1=2.7182818…$. أكتب برنامج يحسب جدول يبين n! و S_n والقيمة المطلقة والقيمة النسبية للخطأ عند قيم n التالية $1\le n\le 20$. إشرح النتائج التى حصلت عليها.

٤- فى التعامل الإحصائى مع البيانات نحتاج فى العادة لحساب المتوسط $\bar{x}$ والانحراف المعيارى S^2 كما يلى:

$$\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i \quad \text{و} \quad S^2=\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2$$

حيث $x_1, x_2, …x_n$ هى البيانات المعطاة. عندما تكون n كبيرة، n=10000 مثلا، فإنه من السهل كتابة S^2 على الصورة التالية:

$$S^2=\frac{1}{n}\sum_{i=1}^{n}x_i^2-\bar{x}^2$$

(أ) أى الطريقتين لحساب S^2 تتوقع أنها ستعطى نتائج أكثر دقة؟

(ب) أكتب برنامج ماتلاب واحسب S^2 بالطريقتين للتأكد من إجابتك السابقة.

٥- الصورة العامة لجذرى أى معادلة من الدرجة الثانية على الصورة $ax^2+bx+c=0$ حيث $a\neq 0$ هى:

$$x_1=\frac{-b+\sqrt{b^2-4ac}}{2a} \quad \text{و} \quad x_2=\frac{-b-\sqrt{b^2-4ac}}{2a}$$

عندما تكون b^2 أكبر كثيرا جدا من 4ac فإن الجذر سيعطى كمية قريبة من b، وعلى ذلك سيكون البسط فى الجذر x_1 عبارة عن فرق بين كميتين متقاربتين ولذلك يمكن أن يحدث خطأ الإلغاء فى حساب x_1، وذلك بالطبع عندما تكون b موجبة، وفى هذه الحالة لن يحدث الخطأ فى حساب الجذر x_2. بنفس الطريقة عندما تكون b سالبة فإن الخطأ من الممكن أن يحدث فى الجذر x_2 وليس فى الجذر x_1. يمكن استخدام صورتان بديلتان لكل من x_1 و x_2 للتخلص من هذا الخطأ كما يلى:

$$x_1=\frac{-2c}{b+\sqrt{b^2-4ac}} \quad \text{و} \quad x_2=\frac{-2c}{b-\sqrt{b^2-4ac}}$$

(أ) أثبت الصورتين البديلتين لكل من x_1 و x_2.

(ب) أوجد الجذرين x_1 و x_2 للمعادلة $x^2+62.10x+1=0$ باستخدام الصورتين الأصلية والبديلة واحسب الخطأ النسبى فى كل جذر فى كل حالة إذا علمت أن القيمة الحقيقية للجذرين هى بالتقريب $x_1=-0.01610723$ و $x_2=-62.08390$.

٦- استخدم الصورتين الأصلية والبديلة لحساب الجذرين لكل واحدة من المعادلات التالية مع حساب الخطأ النسبى بين نتيجة الصوورتين:

(أ) $\frac{1}{3}x^2-\frac{123}{4}x+\frac{1}{6}=0$

(ب) $\frac{1}{3}x^2+\frac{123}{4}x-\frac{1}{6}=0$

(ت) $1.002x^2-11.01x+0.01265=0$

(ث) $1.002x^2 + 11.01x + 0.01265 = 0$

٧- افترض الدالة $f(x) = 1.01x^5 - 5.262x^3 - 0.01732x^2 + 0.8389x - 1.912$.

(أ) إحسب الدالة f(2.279) مستخدما التقريب لأربع خانات عشرية.

(ب) استخدم الصورة البديلة التالية للدالة f(x) واحسب f(2.279) مستخدما أربع خانات عشرية أيضا:

$f(x)=(((1.013x^2 - 5.262)x - 0.01732)x + 0.8389)x) - 1.912$

(ت) احسب الخطأ المطلق والنسبي بين النتيجة فى الجزأين (أ) و (ب).

٨- تقريب كثيرة حدود ماكلورين من الدرجة الخامسة للدالتين e^{2x} و e^{-2x} هما:

$$P5(x) = \left(\left(\left(\left(\frac{4}{15}x + \frac{2}{3}\right)x + \frac{4}{3}\right)x + 2\right)x + 2\right)x + 1$$

و

$$\overline{P5}(x) = \left(\left(\left(\left(\frac{4}{15}x + \frac{2}{3}\right)x + \frac{4}{3}\right)x + 2\right)x + 2\right)x + 1$$

(أ) إحسب $e^{-0.98}$ مستخدما الدالة $\overline{P5}$ (0.49) والتقريب لأربع خانات عشرية.

(ب) إحسب الخطأ المطلق والنسبى للتقريب الحادث فى الجزأ (أ).

(ت) إحسب $e^{-0.98}$ مستخدما العلاقة 1/P5(0.49) مع التقريب لأربع خانات عشرية.

(ث) إحسب الخطأ المطلق والنسبى للنتيجة التى حصلت عليها فى الجزء (ت).

٩- استخدم تقريب القطع عند ثلاث خانات عشرية لحساب المجموع $\sum_i^{10} 1/i^2$ مستخدما أولا الطريقة التالية:

$\frac{1}{1} + \frac{1}{4} + \cdots + \frac{1}{100}$ ثم مستخدما ثانيا الطريقة التالية:

$\frac{1}{100} + \frac{1}{81} + \cdots + \frac{1}{1}$

أى الطريقتين أكثر دقة، ولماذا؟

١٠- الرقم e يتم تحديده بأنه $e = \sum_{n=0}^{\infty} 1/n!$، حيث n! هو مضروب n عندما n≠0 و 0!=1. إستخدم تقريب القطع عند أربع خانات عشرية واحسب الخطأ المطلق والنسبى لكل من التقريبات التالية:

(أ) $\sum_{n=0}^{5} 1/n!$

(ب) $\sum_{j=0}^{5} 1/(5-j)!$

(ت) $\sum_{n=0}^{10} 1/n!$

(ث) $\sum_{j=0}^{10} 1/(10-j)!$

الفصل ٤

حل المعادلات غير الخطية ذات المتغير الواحد

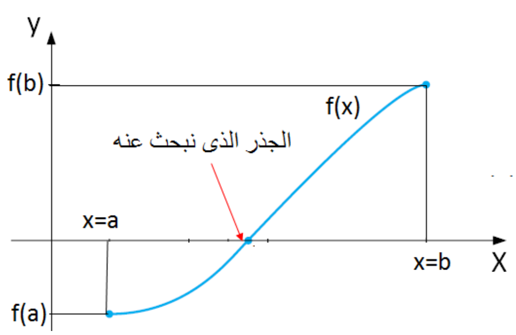
الفصل ٤

حل المعادلات غير الخطية ذات المتغير الواحد

سنتناول فى هذا الفصل الطرق المختلفة لحل المعادلات غير الخطية ذات المتغير الواحد التى على الصورة $f(x)=0$. هذه الحلول تسمى فى معظم الأحوال الجذور أو الأصفار لهذه المعادلة حيث بالتعويض بأحد هذه الجذور أو الأصفار فى التعبير $f(x)$ فإن النتيجة تكون صفرا. المعادلة $f(x)=0$ يكون لها فى العادة أكثر من جذر أو حل، والطرق التى سنقدمها هنا تبحث عن حل واحد معين فى نطاق معين للمتغير x. على الرغم من قدم مشكلة البحث عن جذور مثل هذه المعادلات إلا إنها مازالت نشطة حتى الآن ولازالت تجرى فيها الأبحاث.

٤-١ طريقة التنصيف

هذه الطريقة هى الأكثر بدائية فى الطرق التى سنقدمها هنا وأحيانا تسمى طريقة البحث الثنائى. تستخدم هذه الطريقة لتحديد حل أو صفر للمعادلة $f(x)=0$، بدقة معينة وحسبما تسمح دقة الحاسب المستخدم، فى مدى معين من المتغير x وليكن المدى $[a,b]$ مع ضرورة أن تكون الدالة $f(x)$ مستمرة فى هذا المدى وأن كل من $f(a)$ و $f(b)$ يكون لهما إشارتان مختلفتان. بمعنى أن النقطتين a و b تقعان على جانبين مختلفين من الجذر أو الصفر الذى نبحث عنه.

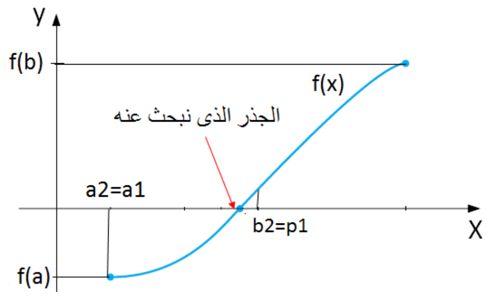


شكل ٤-١ الدالة $f(x)$ لها جذر فى المدى $[a, b]$

شكل ٤-١ يبين خطوات تنفيذ طريقة التنصيف للوصول إلى جذر المعادلة $f(x)=0$ وهى كما يلى:

١- كما ذكرنا فإن هذه الطريقة تفترض أن الجذر الذى نبحث عنه لهذه المعادلة يقع فى المدى $a \le x \le b$ أو بمعنى آخر أن النقطة $x=a$ تقع على يسار الجذر والنقطة $x=b$ تقع على يمين الجذر الذى نبحث عنه وهو نقطة تقاطع المنحنى $f(x)$ مع المحور x كما فى شكل ٤-١. وللتأكد من ذلك نقوم بالتعويض عن $x=a$ و

x=b فى المعادلة f(x) لنوجد كل من f(a) و f(b) حيث يجب أن يكون كل منهما له إشارة عكس الآخر، بمعنى إذا كانت f(a) سالبة فلابد أن تكون f(b) موجبة والعكس صحيح كما فى شكل٤-١.



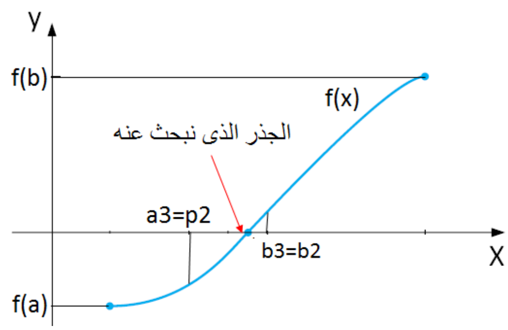
شكل ٤-٢ الخطوة ٤ و ٥ فى خوارزيم التنصيف

٢- سنبدأ الطريقة بوضع a1=a و b1=b، ثم نحسب نقطة المنتصف بين a1 و b1 كما فى المعادلة (٤- ١):

$$p1 = a1 + \frac{b1-a1}{2} \qquad (٤-١)$$

حيث p1 هى نقطة المنتصف كما فى شكل ٤-٢.

٣- نعوض بقيمة p1 فى التعبير f(x) لنحسب f(p1). إذا كانت f(p1)=0 فإن p1 تعتبر هى الجذر الذى نبحث عنه، وننهى الخوارزيم عند هذا الحد.



شكل ٤-٣ الخطوتين ٧ و ٨ للتنصيف الثالث

٤- إذا كانت f(p1) موجبة فإن ذلك يعنى أن النقطة p1 تقع على يمين الجذر ويكون الجذر فى هذه الحالة واقعا فى المدى [a1, p1] كما فى شكل ٤-٢.

٥- نقوم بتنصيف المدى [a1, p1] بجعل b2=p1 و a2=a1 واستخدام المعادلة (٤-٢):

$$p2 = a2 + \frac{b2-a2}{2} \qquad (٤-٢)$$

٦- نقوم بالتعويض عن p2 فى التعبير f(x) لنحسب f(p2). إذا كانت f(p2)=0 فإن p2 تعتبر هى الجذر الذى نبحث عنه، وننهى الخوارزيم عند هذا الحد.

٧- إذا كانت f(p2) موجبة فإن ذلك يعنى أن النقطة p2 تقع على يمين الجذر ويكون الجذر فى هذه الحالة لايزال واقعا فى المدى [a1, p2]. أما إذا كانت f(p2) سالبة فإن ذلك يعنى أن النقطة p2 تقع على يسار الجذر ويكون الجذر فى هذه الحالة واقعا فى المدى [p2,b2].

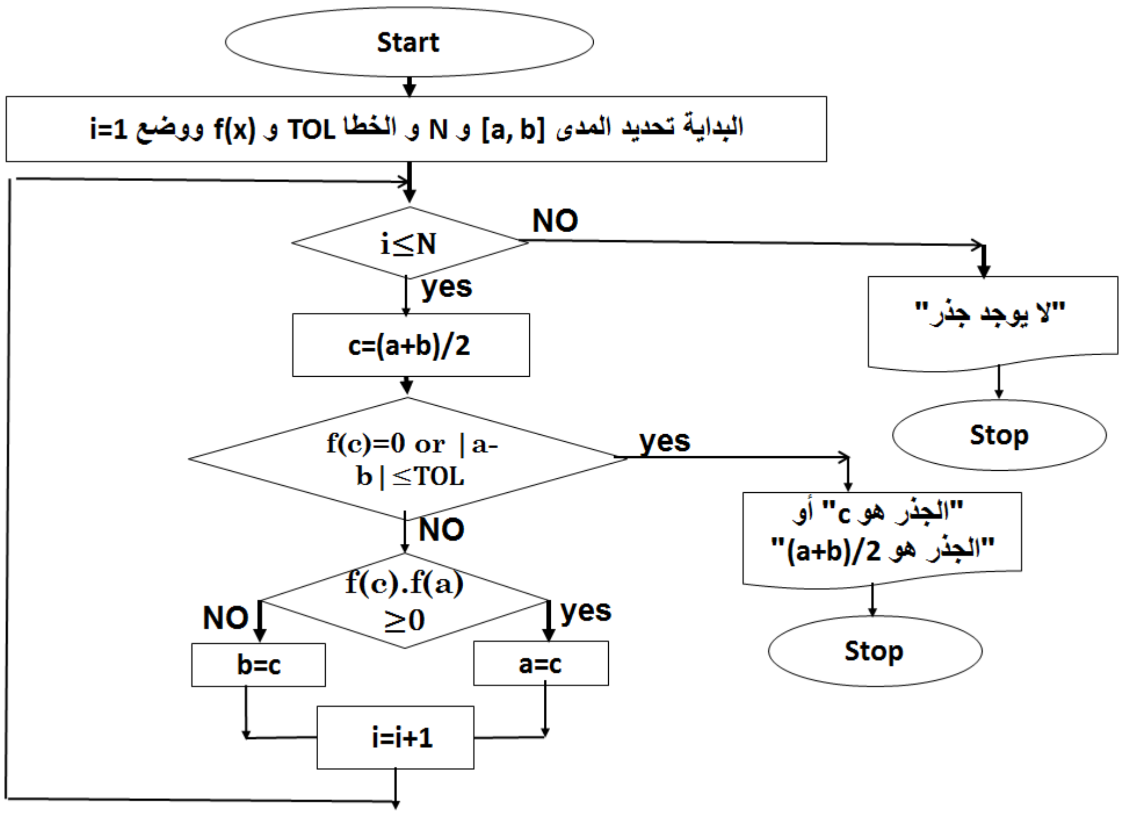
٨- نقوم بتنصيف المدى [p2, b2] بجعل a3=p2 و b3=b2 ونستخدم المعادلة (٤- ٣) فى حساب نقطة المنتصف الجديدة p3 وكما فى شكل ٤-٣.

$$p3 = a3 + \frac{b3-a3}{2} \qquad (٤-٣)$$

٩- على ضوء f(p3) سنقرر كما سبق إما أن ننهى الخوارزيم إذا كانت f(p3)=0 ويكون الجذر هو x=p3، أو أن f(p3) موجب فنجعل b4=p3 و a4=a3 أو أن f(p3) سالبة فنجعل a4=p3 و b4=b3.

١٠- نستمر فى عملية التنصيف هكذا إلى أن يصبح الفرق بين جذرين p_n و p_{n-1} أقل من نسبة خطأ معين تكون هى دقة تحديد هذا الجذر حيث عندها نوقف الخواريزم ويكون الجذر هو p_n.

شكل ٤-٤ يبين مخطط تدفق لطريقة التنصيف السابقة. نلاحظ أنه فى خطوة البداية يتم تحديد العوامل والثوابت التى سيتم استخدامها فى الخواريزم مثل الدالة f(x)، والمدى [a, b] المفترض أن يقع فيه الجذر، و N وهى أكبر عدد من محاولات التنصيف التى إذا تخطاها البرنامج نقول أنه لن يتم الوصول إلى حل أو جذر كأن يكون البرنامج قد دخل فى حلقة مغلقة مثلا وفى هذه الحالة يجب إنهاء البرنامج برسالة "لا يوجد جذر". يتم هنا أيضا تحديد مقدار الخطأ TOL فى حساب المدى الذى يقع فيه الجذر [a, b] بحيث إذا وصل الخواريزم إلى هذا الخطأ TOL يكون قد وصل إلى الجذر المطلوب بهذه النسبة من الخطأ.



شكل ٤-٤ مخطط التدفق لخواريزم التنصيف للحصول على جذر أى معادلة غير خطية فى متغير واحد

بعد ذلك يدخل الخواريزم فى حلقة على المتغير i التى تبدأ من i=1 حتى i≤N. فى داخل هذه الحلقة يتم حساب نقطة المنتصف c=(a+b)/2، وبعدها يتم تحديد إذا كانت نقطة المنتصف تلك فى جانب النقطة a أم فى جانب النقطة b عن طريق حاصل الضرب f(c).f(a). فإذا كان حاصل الضرب موجب فإن ذلك يعنى أن f(c) و f(a) لهما نفس

الإشارة، أى أن نقطة المنتصف تقع فى ناحية النقطة a وفى هذه الحالة نجعل a=c، وتظل b كما هى. إذا كان حاصل ضرب f(c) فى f(a) سالب فإن ذلك يعنى أن نقطة المنتصف c تقع ناحية النقطة b وفى هذه الحالة سنجعل b=c ونترك a كما هى. بعد ذلك نجمع واحد على متغير الحلقة لنبدأ دورة جديدة. يظل تنفيذ الحلقة إلى أن يخرج من أحد المخرجين الموضحين فى شكل ٤-٤.

البرنامج التالى يوضح مقترحا لتنفيذ هذا الخوارزم والتعليق على أوامره المختلفة بعد البرنامج مباشرة:

```
1- %bisection method to find a root for nonlinear equation in a single
   var.
2- f1 = input('Enter The Function: ' , 's') ;%for example x^2-2
3- f = inline(f1) ;
4- y = input('Enter The Interval: ' ) ;%for example [0 2]
5- n = input('Enter the Number of iterations: ') ;
6- t = input('Enter the Tolerance: ') ;
7- a = y(1) ; b = y(2) ;
8- z = f(a) ; v = f(b) ;
9- if z*v > 0
10- disp('The Root Not Found, wrong interval!')
 11-  break
12-      else
13-      i = 0 ;
14-      while i <= n
15-          c = (a+b)/2 ;
16-          if f(c) == 0 || abs((b-a)/2) <= t
17-           disp([' The Root = ' num2str(c)])
18-           break
19-          elseif sign(f(c)) == sign(f(a))
20-              a = c ;
21-          else
22-              b = c ;
23-          end
24-          i = i+1 ;
25-         end
26-         if i>n
27-             disp ('The Root Not Found, need more iterations!');
28-         end
29-      end
```

السطر الأول الذى يبدأ بالعلامة % يعتبر تعليقا أو سطرا غير قابل للتنفيذ مكتوب فيه نصا يفيد أن هذا البرنامج هو لتنفيذ طريقة التنصيف لإيجاد جذر معادلة غير خطية فى متغير واحد.

يأتى بعد ذلك عدد من أوامر الإدخال input التى تستخدم لإدخال ثوابت البرنامج. بالنسبة للأمر

```
f1 = input('Enter The Function: ' , 's');%for example x^2-2
```

مابين علامتى التنصيص 'Enter The Function: ' عبارة عن نص يكتب كما هو حيث يخبر المستخدم بأن يدخل الدالة f(x) المراد إيجاد جذرها ولقد تم إعطاء مثالا على ذلك كتعليق بعد العلامة % فى نهاية السطر. المثال هنا هو x^2-2. أى أن المعادلة f(x) ستكون على الصورة f(x)=x2-2=0، أى أننا نريد إيجاد جذر 2. المعامل 's' فى الأمر

input السابق يخبر ماتلاب بأن يعتبر ما سيدخله المستخدم عبارة عن نص strimg ولا يعتبره معادلة فيقوم بإجرائها. نص المعادلة الذى سيقوم المستخدم بكتابته سيضعه ماتلاب فى المتغير f1. أمر الإدخال:

```
y = input('Enter The Interval: ' ) ;%for example [0 2]
```

يطلب من المستخدم أن يدخل المدى الموجود فيه الجذر ويتم إدخاله فى صورة صف فى مصفوفة مثل [0 2]، حيث بمجرد إدخاله يضعه الماتلاب فى المتغير y. أمر الإدخال التالى يطلب من المستخدم أن يدخل عدد حلقات تنفيذ الخواريزم ويضعه فى المتغير n، وأمر الإدخال التالى يطلب من المستخدم أن يدخل الخطأ الذى ستقف عنده المحاولات ويضعه فى المتغير t.

فى السطر الثالث هناك الأمر ;f=inline(f1)، سيجعل النص الموجود فى المتغير f1 معادلة قابلة للحساب ويضعها فى المتغير الجديد f. أى أن f بعد هذا الأمر ستصبح f=x^2-2.

فى السطر ٧: a=y(1) و b=y(2) أى أن a تساوى العنصر الأول من الصف y و b تساوى العنصر الثانى. بعد ذلك فى السطر ٨ تم حساب الدالة f(x) عند كل من a و b عن طريق حساب z=f(a) و v=f(b). قبل الدخول فى الحلقة لابد من التأكد من أن النقطتين a و b على جانبين مختلفين من الجذر، أو بمعنى آخر لابد من التأكد من أن كل من z=f(a) و v=f(b) لهما إشارتين مختلفتين، وقد تم ذلك فى الأسطر ٩ و ١٠ و ١١ حيث تم ضرب z*v والسؤال إن كانت النتيجة أكبر من الصفر (if z*v>0) حيث فى هذه الحالة يكون كل من z و v لهما نفس الإشارة ويجب الخروج من البرنامج حيث أن شرط f(a) و f(b) أحدهما سالبة والأخرى موجبة غير محقق ولذلك يتم الخروج من البرنامج بالأمر ;break. إذا كانت كل من z و v تحققان هذا الشرط فإن البرنامج سيستمر فى التنفيذ، ويدخل حلقة التنفيذ بوضع i=0، ويستمر البرنامج فى تنفيذ الحلقة طالما أن i≤n. لاحظ أن i يتم زيادتها بمقدار 1 فى السطر ٢٤ قبل نهاية الحلقة وقبل الدخول فى الدورة الجديدة من الحلقة. فى كل حلقة يتم حساب نقطة المنتصف c كما فى السطر ١٥، والسؤال إذا كانت c هى الجذر أو إذا كان المدى (a-b)/2 أقل من الخطأ t أو يساويه كما فى السطر ١٦ حيث فى هذه الحالة يكون قد تم العثور على الجذر ويمكن الخروج من البرنامج وكتابة الجذر. كتابة الجذر تتم باستخدام الأمر disp(['The Root=', num2str(c)]) الذى يكتب النص 'The Root=' ثم يكتب من بعده قيمة الجذر c، ولكن حيث أن c عبارة عن قيمة حسابية بينما يتم هنا كتابة نصوص فكان لابد من تحويل c من قيمة حسابية إلى سلسلة أحرف أو نص باستخدام الأمر num2str(c) الذى يحول c من قيمة حسابية إلى سلسلة أحرف لكى تعرض بالأمر disp. من الأوامر المستخدمة فى هذه الحلقة أيضا الأمر sign(f(c)) حيث الدالة sign() هى دالة الإشارة، فإذا كانت f(c) موجبة فإن sign(f(c)) تساوى واحد، وإذا كانت f(c) تساوى صفر فإن sign(f(c)) تساوى صفر أيضا، وأما إذا كانت f(c) سالبة فإن sign(f(c)) تساوى 1-. لذلك فإن السطر ١٩ يسأل هل إشارة f(c) تساوى

إشارة $f(a)$. فى النهاية إذا انتهت الحلقة i $(i>n)$ ولم يتم الوصول إلى الجذر بالدقة المطلوبة فإن البرنامج يخرج من الحلقة ويعرض الرسالة disp('The Root not Found, Need More Iteration!') وينتهى البرنامج.

الآن سنقوم بتنفيذ البرنامج على الدالة $f(x)=x^2-2$ حيث سنحصل على مايلى:

```
>> bisection
Enter The Function: x^2-2
Enter The Interval: [0 2]
Enter the Number of iterations: 20
Enter the Tolerance: 0.0005
The Root = 1.4146
```

لقد تم وضع إجاباتنا على أوامر الإدخال باللون الأزرق، حيث فى النهاية كانت نتيجة البرنامج هى أن الجذر يساوى 1.4146 وهو جذر 2.

كمثال آخر لإثبات صحة خواريزم التنصيف السابق سنفترض الدالة $f(x)=x^2-2x-3=(x-3)(x+1)=0$ التى لها جذران أحدها عند $x=3$ والآخر عند $x=-1$ وسننفذ البرنامج للبحث عن الجذر عند $x=-1$ كما يلى:

```
>> bisection
Enter The Function: x^2-2*x-3
Enter The Interval: [-1.5 -0.5]
Enter the Number of iterations: 20
Enter the Tolerance: 0.0005
 The Root = -1
```

من الملاحظات المهمة عند وضع المدى الذى نبحث فيه عن الجذر يجب أن نجعل هذا المدى قليلا أو ضيقا بقدر الإمكان لتقليل زمن تنفيذ الخواريزم أو الوصول للجذر المطلوب. فمثلا فى مثال الدالة $f(x)=x^2-2=0$ والتى تبحث عن جذر الرقم 2 كنا نبحث عن هذا الجذر فى المدى [0 2]. بالطبع لو جعلنا المدى [1 2] فإن البرنامج سيصل إلى الجذر أسرع. ننصح القارىء بأن يأخذ نسخة من البرنامج ويضعها مباشرة فى ماتلاب ويحاول اختبار البرنامج بثوابت ومعاملات مختلفة.

للتأكد من أن طرفى المدى الذى نبحث فيه عن الجذر $f(a)$ و $f(b)$ يكون أحدهما موجب والآخر سالب يمكننا استخدام حاصل ضرب الدالتين $f(a).f(b)$ بحيث يكون سالب. المشكلة مع حاصل الضرب أنه من الممكن أن تكون كل $f(a)$ و $f(b)$ كبيرة جدا وبالتالى فإن حاصل الضرب سيكون أكبر وأكبر ولذلك من الممكن أن يحدث فيضان فى نتيجة الضرب، أى تكون النتيجة أكبر من أكبر رقم يمكن للحاسب أن يتعامل معه. لتجنب هذه المشكلة فإننا نستخدم حاصل ضرب دالة الإشارة sign(f(a)).sign(f(b)) حيث الدالة sign() تكون إما 1 أو 0 أو -1 على حسب إذا

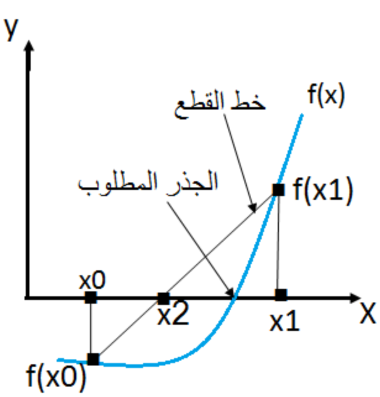
كان معامل الدالة موجب أو سالب أو صفر كما رأينا مسبقا. لذلك لا يمكن أن يحدث فيضان فى هذه الحالة لأن ناتج الضرب سيكون 1 أو 0 أو 1-.

يحتوى ماتلاب على دالتين لإيجاد جذور أى معادلة مباشرة يمكن استخدامهما للتحقق من صحة البرنامج الذى تكتبه. الدالة الأولى هى fzero(FUN, X0) حيث هذه الدالة تبحث عن الصفر القريب من النقطة X0 للدالة FUN. حاول تجربة هذه الدالة على $f(x)=x^2-2$ وعلى $f(x)=x^2-2x-3$ اللتين حسبناهما مسبقا كما يلى:

```
>> fzero('x^2-2',1)
ans =
   1.4142
>> fzero('x^2-2*x-3', -2)
ans =
   -1
```

الدالة الثانية التى تحسب الجذور بصورة جاهزة فى ماتلاب هى الدالة roots (C) التى تحسب جذور المعادلة أو كثيرة الحدود التى معاملاتها معطاة فى المتجه C. مثلا الدالة $f(x)=x^2-2x-3$ معاملاتها هى c=[1 -2 -3] والدالة $f(x)=x^2-2$ معاملاتها هى a=[1 0 -2] ولذلك يمكن حساب جذورهما مباشرة بالدالة roots() كما يلى:

```
>> c=[1 -2 -3];
>> r=roots(c)
r =
   3.0000
  -1.0000
>> a=[1 0 -2];
>> r=roots(a)
r =
   1.4142
  -1.4142
```

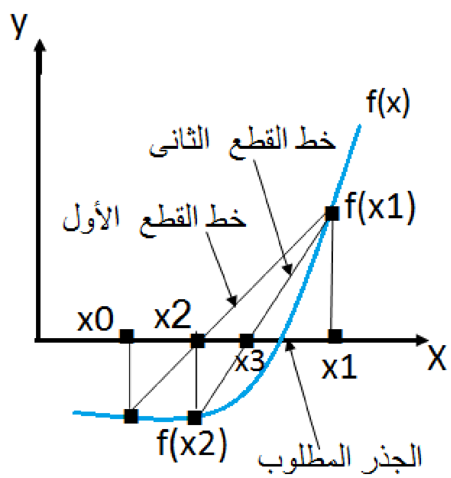


شكل ٤-٥ الخطوة الأولى فى طريقة خط القطع لإيجاد جذر أى معادلة غير خطية فى متغير واحد

لاحظ أن الدالة roots () تعطى كل جذور المعادلة أو كثيرة الحدود بخلاف الدالة fzero() أو الخوارزم الذى درسناه فإنه يبحث عن جذر للمعادلة المعطاة قريبا من نقطة معينة أو محصورا فى مدى معين.

بعد أن رأينا الدالتين fzero() و roots() ربما يتبادر إلى ذهننا السؤال التالى: لماذا نتعلم طريقة التنصيف والطرق الأخرى التى سندرسها فيما بعد والماتلاب يحل الأمور بهذه السهولة؟ الإجابة فى الحقيقة بسيطة وهى أنه ليس كلنا يعرف ماتلاب أو يجيد استخدامه ولذلك فإنه بمعرفتنا لخوارزمات إيجاد الجذور يمكننا أن نكتب هذه الخوارزمات بأى لغة برمجة نعرفها أو نجيدها، وليس بالضرورة أن نشترى ماتلاب بالكامل لكى نستخدمه فى إيجاد جذور هذه معادلة.

خواريزم التصيف والخواريزمات التى سندرسها فيما بعد تتعامل فقط مع الجذور الحقيقية، أما الجذور المركبة فإن هذه الخواريزمات غير معدة لها. ولكن الدالة roots() يمكنها التعامل مع الجذور المركبة أيضا.



شكل ٤-٦ المحاولة الثانية للاقتراب من الجذر بطريقة الخط القاطع

٤-٢ طريقة الخط القاطع لحساب الجذر

تسير طريقة الخط القاطع secant تبعا للخطوات التالية التى تشبه إلى حد كبير طريقة التنصيف السابقة:

١- تبدأ الطريقة بتحديد نقطتين x0 و x1 أحدهما على يسار الجذر والأخرى على يمينه كما فى شكل ٤-٥.

٢- بدلا من حساب نقطة المنتصف بين x0 و x1 كما فى طريقة التنصيف السابقة فإننا نحسب f(x0) و f(x1)

وبذلك يصبح لدينا نقطتان على منحنى الدالة f(x) إحداثياتهما هما [x0,f(x0)] و [x1, f(x1)]. نقوم بالتوصيل بين هاتين النقطتين بخط مستقيم كما فى شكل ٤-٥، وهذا الخط سيقطع المحور x عند النقطة x2. فى الحقيقة أن ما حدث فى هذه الخطوة أننا قربنا المنحنى f(x) بالخط المستقيم بين هاتين النقطتين، وبذلك فإن النقطة x2 ستعتبر أول تقريب لجذر المعادلة الذى نبحث عنه وسنحاول الاقتراب منه أكثر فى الخطوات السابقة. معادلة هذا الخط المستقيم يمكن كتابتها كما يلى:

$$y = f(x1) + \frac{f(x1)-f(x0)}{x1-x0}(x - x1) \qquad (٤-٤)$$

بوضع y=0 فى المعادلة (٤-٤) نحصل على x2 التى تمثل الجذر المقترح كما يلى:

$$x2 = x1 - f(x1)\frac{x1-x0}{f(x1)-f(x0)} \qquad (٤-٥)$$

٣- بمعرفة x2 من المعادلة (٤- ٥) يمكن التعويض بها فى الدالة f(x) لنحصل على f(x2). الآن أصبح لدينا نقطة جديدة على المنحنى f(x) هى النقطة [x2, f(x2)]. هذه النقطة الجديدة [x2, f(x2)] مع النقطة [x1, f(x1)] يمكن التوصيل فيما بينهما كما فى شكل ٤-٦ فى خط مستقيم يقطع المحور x عند النقطة x3 التى تعتبر تقريبا جديدا للجذر الذى نبحث عنه. معادلة الخط المستقيم الجديد يمكن كتابتها كما يلى:

$$y = f(x2) + \frac{f(x2)-f(x1)}{x2-x1}(x - x2) \qquad (٤-٤)$$

بوضع y=0 فى المعادلة (٤- ٥)نحصل على x3 التى تمثل الجذر الجديد المقترح كما يلى:

$$x3 = x2 - f(x2)\frac{x2-x1}{f(x2)-f(x1)} \qquad (٥-٤)$$

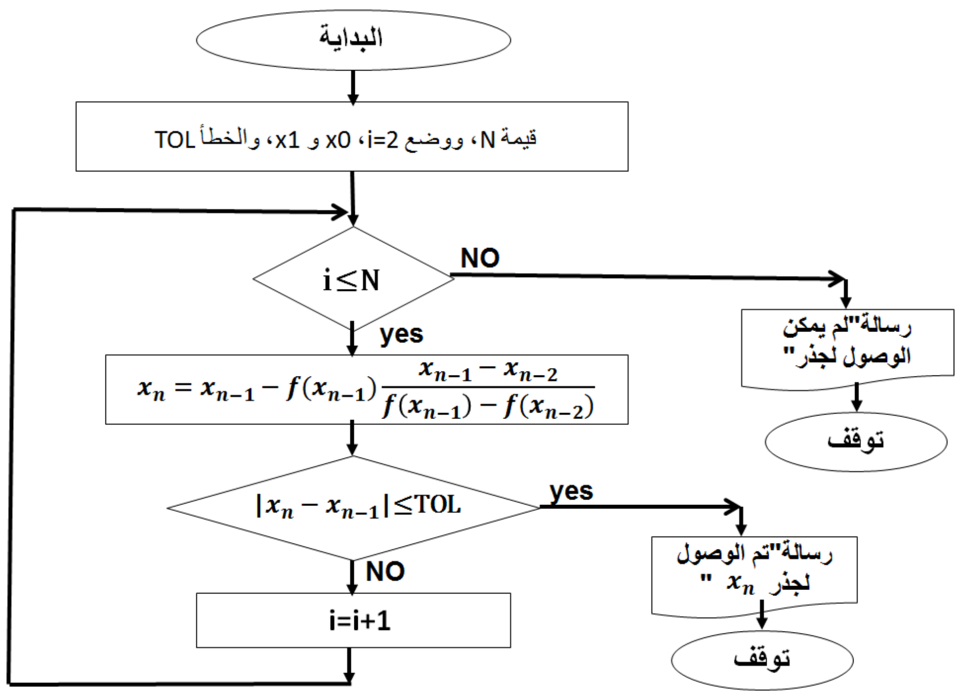
٤- بنفس الطريقة نوجد النقطة الجديدة [x3, f(x3)]، ثم نوجد معادلة الخط الجديد الذى يصل بين النقطتين [x1, f(x1)] و [x3, f(x3)]، ونوجد تقاطع هذا الخط مع المحور x لنجد النقطة x4 المقترح الجديد للجذر.

٥- معنى ذلك أنه يمكننا كتابة معادلة النقطة xn التى تمثل المقترح رقم n للجذر كما يلى:

$$x_n = x_{n-1} - f(x_{n-1})\frac{x_{n-1}-x_{n-2}}{f(x_{n-1})-f(x_{n-2})} \qquad (٦-٤)$$

٦- نستمر فى حساب الجذر تبعا للمعادلة (٤- ٦) حتى نصل إلى الوضع الذى يكون عنده الفرق بين xn و xn-1 أقل من قيمة خطأ أو دقة يتم تحديدها من البداية، حيث عندها نخرج من الحلقة ونعلن أن الجذر هو xn، ونوقف البرنامج.

شكل ٤-٧ يبين مخطط تدفق لخوارizم الخط القاطع لإيجاد جذر أى معادلة غير خطية فى متغير واحد.



شكل ٤-٧ مخطط تدفق لخوارizم الخط القاطع حساب جذر أى معادلة غير خطية ذات متغير واحد

من مميزات هذه الطريقة أنها أسرع فى الوصول إلى الجذر من طريقة التنصيف السابقة، ولكن عيبها أن منطقة البحث عن الجذر ليست محدودة كما فى طريقة التنصيف ولذلك فإن فرصة الخروج بدون حل تكون أكبر فى هذه الحالة، وفى هذه الحالة فإن تغيير نقط البداية قد يصحح ذلك.

البرنامج التالى مجرد مقترح لبرنامج باستخدام ماتلاب لتنفيذ خواريزم الخط القاطع لإيجاد جذر أى معادلة غير خطية فى متغير واحد، بالطبع يمكن للقارىء عمل البرنامج الخاص به بأى لغة أخرى إن أراد.

```
%secant method to find a root of a nonlinear equation
f1 = input('Enter The Function: ' , 's') ;%for example x^4-2
f = inline(f1) ;
x(1) = input('Enter The 1st Point: ' ) ;%for example 0
x(2) = input('Enter the 2nd point: ' ) ;%for example 3
n = input('Enter the Number of iterations: ') ;
t = input('Enter the Tolerance: ') ;
if f(x(1))*f(x(2)) > 0
    disp('The Root Not Found, wrong interval!')
          break
      end
       for i = 2 : n
              x(i+1)=x(i)-(f(x(i))/(f(x(i))-f(x(i-1))))*(x(i)-x(i-1));
             if abs(x(i) - x(i-1)) < t
                c = x(i+1) ;
                 disp(['The Root = ' num2str(c)])
                 break
             end
             if i== n
                 disp('The Root Not Found, need more iterations!');
             end
         end
```

جميع أوامر البرنامج تم شرحها فى البرنامج السابق ولا حاجة لإعادة شرحها هنا.

الآن سنطبق البرنامج على نفس المثالين المستخدمين مع البرنامج السابق:

أولا مع الدالة $f(x)=x^2-2$ كانت نتيجة التنفيذ كما يلى:

```
>> secant
Enter The Function: x^2-2
Enter The 1st Point: 0
Enter the 2nd point: 2
Enter the Number of iterations: 20
Enter the Tolerance: 0.0005
The Root = 1.4142
```

ثانيا مع الدالة $f(x)=x^2-2x-3$ كانت نتيجة التنفيذ كما يلى:

```
>> secant
Enter The Function: x^2-2*x-3
Enter The 1st Point: -1.5
Enter the 2nd point: -0.5
Enter the Number of iterations: 30
Enter the Tolerance: 0.0005
```

The Root = -1

ننصح القارىء بأن يضيف إلى برنامجى طريقة التنصيف وطريقة الخط القاطع أن يعطى البرنامج قيمة n (عدد الدورات أو عدد المحاولات) التى وصل عندها البرنامج إلى الجذر المطلوب والمقارنة لترى أى البرنامجين أسرع.

مثال ٤-١

نفذ ثلاث خطوات لإيجاد جذر المعادلة $f(x)=x^3-4$ الموجود فى المدى [1 3]. إحسب الخطأ فى كل خطوة.

سنوجد أولا جذور المعادلة الثلاثة الحقيقية باستخدام دالة ماتلاب الجاهزة roots() لنستخدمها فى التحقق من الصحة كما يلى:

```
>> a=[1 0 0 -4];
>> z=roots(a)
z =
 -0.7937 + 1.3747i
 -0.7937 - 1.3747i
  1.5874 + 0.0000i
```

حيث a هى معاملات كثيرة الحدود: $f(x)=x^3+0x^2+0x-4$. من الواضح من ذلك أن هذه المعادلة لها جذرين مركبين وهذان خارج حدود ما نقدمه فى هذا الكتاب وجذر واحد حقيقى عند $x=1.5874$.

ثانيا باستخدام طريقة التنصيف:

١- المدى الموجود فيه الجذر هو [1 3] لذلك فإن $a=1$ و $b=3$ و $f(a)=1^3-4=-3$ و $f(b)=3^3-4=23$ إذن $f(a).f(b)<0$ وبالتالى فالمدى صحيح.

٢- أول منتصف هو $c=(a+b)/2=2$. وبالتالى $f(c)=2^3-4=4$، أى أن c أقرب إلى b وبالتالى سنجعل $b=c=2$ وستظل $a=1$ كما هى وننتقل إلى الخطوة التالية.

٣- باعتبار أن $a=1$ و $b=2$ فإن المنتصف الجديد سيكون $c=(a+b)/2=1.5$. وبالتالى $f(c)=1.5^3-4=-0.6250$، أى أن c هذه المرة أقرب إلى a وبالتالى سنجعل $a=c=1.5$ وستظل $b=2$ كما هى وننتقل إلى الخطوة التالية.

٤- باعتبار أن $a=1.5$ و $b=2$ فإن المنتصف الجديد سيكون $c=(a+b)/2=1.75$. وبالتالى $f(c)=1.75^3-4=1.3594$، أى أن c تقع هذه المرة أقرب إلى b وبالتالى سنجعل $b=c=1.75$ وستظل $a=1.5$ كما هى وننتقل إلى الخطوة التالية.

٥- إذا اعتبرنا أننا بذلك أتممنا ثلاث خطوات فإن الجذر سيكون عند المنتصف الجديد c=(a+b)/2=1.6250. وبالتالى فإن مقدار الخطأ فى الجذر فى هذه الحالة سيكون: |1.5874-1.6250|=0.0376 وسيكون الخطأ النسبى 0.0376/1.5874x100%=2.37%.

ثالثا باستخدام طريقة الخط القاطع:

٦- فى هذه الحالة x0=1 و x1=3 و f(x0)=-3 و f(x1)=23 بالتعويض فى معادلة تقاطع الخط القاطع مع المحور x نحصل أول جذر كما يلى: $x2 = x1 - f(x1)\frac{x1-x0}{f(x1)-f(x0)} = 3 - 23\frac{3-1}{23-(-3)} =$ 1.2308.

١- الجذر التالى سيكون عند النقطة x3 ويمكن حسابه بنفس الطريقة بمعلومية $f(x2)=1.2308^3-4=-2.1355$:

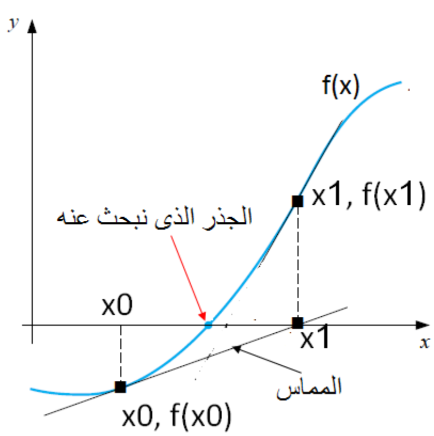
$$x3 = x2 - f(x2)\frac{x2-x1}{f(x2)-f(x1)} = 1.2308 + 2.1355\frac{1.2308-3}{-2.1355-23} = 1.3811$$

٢- الجذر التالى سيكون عند النقطة x4 ويمكن حسابه بنفس الطريقة بمعلومية $f(x3)= 1.3811^3-4=-1.3656$:

$$x4 = x3 - f(x3)\frac{x3-x2}{f(x3)-f(x2)} = 1.3811 + 1.3656\frac{1.3811-1.2308}{-1.3656+2.1355} = 1.6477$$

٣- إذا اعتبرنا أننا بذلك أتممنا ثلاث خطوات فإن الجذر سيكون عند x4. وبالتالى فإن مقدار الخطأ فى الجذر فى هذه الحالة سيكون: |1.5874-1.6477|=0.0603، وسيكون الخطأ النسبى يساوى 0.0603/1.5874x100%=3.80%.

من الواضح أن طريقة التنصيف بعد ثلاث خطوات أعطت الجذر بدقة أفضل من طريقة الخط القاطع، ولكن بالطبع ربما بالاستمرار فى الخطوات للوصول إلى دقة أفضل يتغير الوضع عن ذلك.

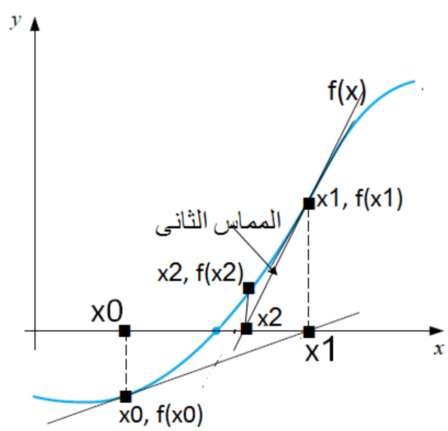


شكل ٤-٨ الخطوة الأولى فى طريقة المماس

٤-٣ طريقة المماس (نيوتن رافسون)

لقد رأينا أن طريقة التنصيف bisection تبدأ بتحديد مدى ابتدائى من نقطتين يحتوى الجذر المطلوب، ثم تقوم بتنصيف المدى المحصور فيه الجذر إلى أنصاف ثم تحسب قيمة الدالة f(x) عند هذه الأنصاف، فإذا كانت هذه الدالة تساوى صفر فهذا هو الجذر المطلوب، وإلا فإنها تستمر فى التنصيف حتى

يصبح المدى الذى يتم البحث فيه عن الجذر أقل من خطأ معين حيث عندها يكون الجذر هو منتصف هذا المدى الأخير. ولقد رأينا أن طريقة الخط القاطع secant تبدأ أيضا بتحديد المدى الابتدائى الذى سيتم البحث فيه عن الجذر بنقطتين، ثم تقوم بتقريب الدالة f(x) إلى خط مستقيم يصل بين هاتين النقطتين، وتحسب نقطة تقاطع هذا الخط المستقيم مع المحور الأفقى، ثم تحسب قيمة الدالة f(x) عند هذه النقطة، وبذلك تتكون نقطة جديدة على المنحنى يتم التوصيل بينها وبين النقطة السابقة بخط مستقيم جديد، وتستمر العملية هكذا إلى أن يتم الوصول إلى الجذر المطلوب. سنرى هنا طريقة المماس tangent أو طريقة نيوتن ورافسون Newton Raphon وهذه الطريقة تبدأ بنقطة ابتدائية واحدة على المنحنى f(x) قريبة من الجذر الذى يتم البحث عنه وليس مدى ابتدائى من نقطتين مثل الطريقتين السابقتين. عند هذه النقطة التى على المنحنى f(x) يتم حساب معادلة المماس للمنحنى f(x)، ثم نوجد نقطة تقاطع هذا المماس مع المحور الأفقى، نوجد نقطة جديدة على المنحنى مقابلة لنقطة التقاطع تلك، ثم نحسب معادلة المماس الجديد للدالة f(x) عند هذه النقطة الجديدة، ثم نوجد تقاطع المماس الجديد مع المحور الأفقى، ثم نوجد نقطة جديدة على المنحنى f(x) مقابلة لنقطة التقاطع مع المحور الأفقى، ونحسب المماس الجديد للدالة f(x) عند هذه النقطة، وهكذا تستمر العملية إلى أن نصل إلى الجذر المطلوب. سنوضح هذه الطريقة بخطوات تفصيلية كما يلى:



شكل ٤–٩ المماس الثانى فى طريقة نيوتن

١– تبدأ الخطوة الأولى بمعرفة النقطة x0 القريبة من الجذر الذى نبحث عنه.

٢– نقوم بالتعويض فى الدالة f(x) بقيمة x0 لنوجد f(x0)، بذلك يكون لدينا نقطة على المنحنى f(x) وهى النقطة [x0, f(x0)] كما هو موضح فى شكل ٤–٨.

٣– من أهم شروط هذه الطريقة أن يكون تفاضل الدالة f(x) عند النقطة x0 ، وهو f'(x0) موجودا. هذا التفاضل f'(x0) يمثل ميل خط المماس عند النقطة x0, f(x0)، وبالتالى يمكن كتابة معادلة هذا الخط بفرض أى نقطة x, y عليه كما يلى:

$f'(x0) = \frac{y-f(x0)}{(x-x0)}$ ومنها يمكن كتابة معادلة هذا المماس كما يلى:

$$y - f(x0) = f'(x0)(x - x0) \qquad (٤-٧)$$

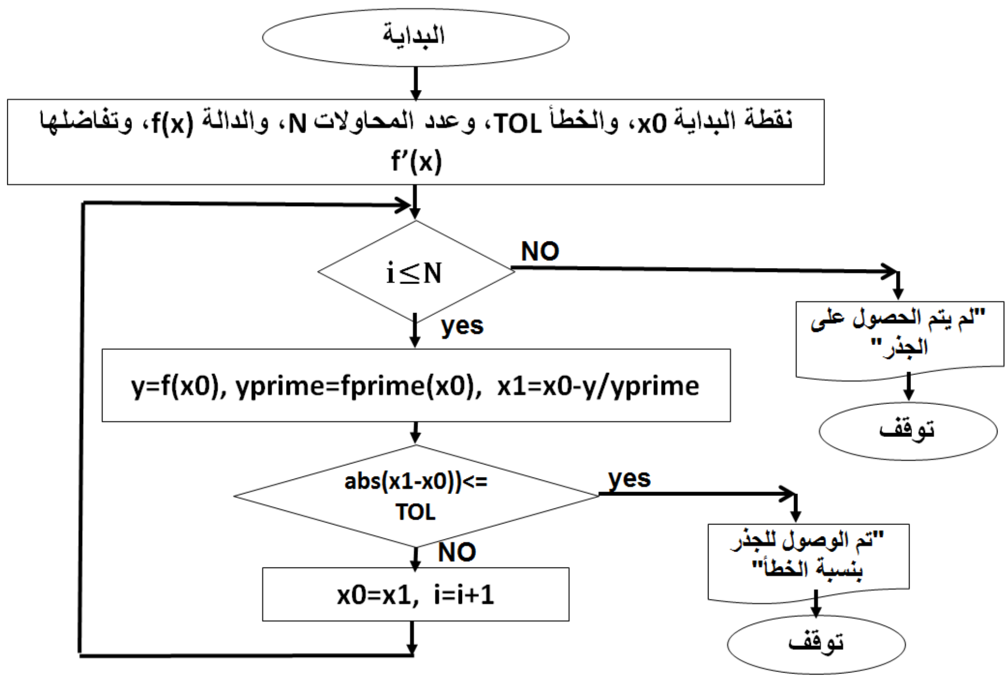
٤– النقطة x1 فى شكل ٤–٨ هى النقطة التى عندها y=0 فى المعادلة (٤– ٧). لذلك بوضع y=0 فى هذه المعادلة نحصل على:

$$x1 = x0 - \frac{f(x0)}{f\prime(x0)} \qquad (٨-٤)$$

٥- بمعرفة x1 نقوم بالتعويض فى الدالة f(x) فنحصل على النقطة x1, f(x1) على هذا المنحنى.

٦- نقوم برسم المماس عند النقطة x1, f(x1) كما فى شكل ٤-٩ حيث يمكن إيجاد نقطة تقاطعه مع المحور الأفقى عند x2 التى يمكن كتابة معادلتها بنفس الطريقة السابقة كما يلى:

$$x2 = x1 - \frac{f(x1)}{f\prime(x1)} \qquad (٩-٤)$$



شكل ٤-١٠ مخطط التدفق لطريقة نيوتن رافسون (المماس) لإيجاد جذر أى معادلة غير خطية فى متغير واحد

٧- نستمر بنفس الطريقة فى إيجاد المماس ثم نقطة التقاطع حيث نلاحظ الاقتراب من الجذر المطلوب، وفى هذه الحالة يمكن كتابة نقطة التقاطع فى الحالة العامة كما يلى:

$$x_{n+1} = x_n - \frac{f(x_n)}{f\prime(x_n)} \qquad (١٠-٤)$$

شكل ٤-١٠ يبين مخطط التدفق لطريقة نيوتن رافسون لحساب جذر أى معادلة غير خطية فى متغير واحد.

البرنامج التالى عبارة تنفيذ لخواريزم المماس (نيوتن رافسون) لإيجاد الجذر لأى معادلة غير خطية فى متغير واحد:

```
1- %Tangent (Newton Raphson) method to find a root of a nonlinear equation
2- f = input('Enter The Function: ' , 's') ;%for example x^2-2
3- f = inline(f);
4- df=input('Enter Function Derivative: ','s');
5- df=inline(df);
6- x0 = input('Enter Starting Point: ');
7- n = input('Enter the Number of iterations: ');
```

```
t = input('Enter the Tolerance: ') ;
for i=1:n
  x1 = x0-(f(x0)/df(x0));
  err = abs(x1-x0);
  if err >= t
    x0=x1;
  else
    disp('The Root =')
    disp(x0)
    break
  end
  end
  if err > t
     disp ('The Root Not Found!')
  end
```

بتنفيذ هذا الخواريزم على المثالين الذين استخدمناهما مع الطريقتين السابقتين، $f(x)=x^2-2$، و $f(x)=x^2-2x-3$ كانت النتيجة كما يلى:

```
>> Newton
Enter The Function: x^2-2
Ente Function Derivative: 2*x
Enter Startung Point: 2
Enter the Number of iterations: 30
Enter the Tolerance: 0.0005
The Root =
   1.4142

>> Newton
Enter The Function: x^2-2*x-3
Enter Function Derivative: 2*x-2
Enter Starting Point: -3
Enter the Number of iterations: 30
Enter the Tolerance: 0.0005
The Root =
  -1.0000
```

مثال ٤–٢

نفذ أول ثلاث خطوات فى طريقة نيوتن رافسون لإيجاد جذر المعادلة $f(x)=x^3-4$ التى سبق أن أجريناها مع طريقتى التنصيف والخط القاطع. كما نعلم من قبل فإن هذه المعادلة لها ثلاث جذور منها اثنين مركبين وجذر واحد حقيقى عند $x=1.5874$، لذلك سنفترض نقطة البداية هى $x0=3$.

١– بالتعويض بقيمة $x0$ فى $f(x)$ نحصل على $f(x0)=23$، تفاضل الدالة $f(x)$ هو $f'(x)=3x^2$ وبالتالى فإن $f'(x0)=27$.

٢– بالتعويض فى المعادلة: $x1=x0-f(x0)/f'(x0)=3-23/27=2.1481$، وهذه أول محاولة.

٣– فى المحاولة الثانية: $x2=x1-f(x1)/f'(x1)= 2.1481-5.9121/13.8430=1.7210$.

٤– فى المحاولة الثالثة: $x3=x2-f(x2)/f'(x2)= 1.7210-1.0973/8.8855=1.5975$.

٥– إذا اعتبرنا أننا بذلك أتممنا ثلاث خطوات فإن الجذر سيكون عند $x3$. وبالتالى فإن مقدار الخطأ فى الجذر فى هذه الحالة سيكون: $|1.5874-1.5975|=0.0101$ وسيكون الخطأ النسبى كما يلى: $0.0101/1.5874x100\%=0.\ 64\%$، وهذا أقل بكثير من الطريقتين السابقتين.

كما رأينا فإن هذه الطريقة تتميز بسرعتها ولكن من عيوبها حساب قيمة التفاضل للدالة وهذه لم تكن موجودة فى الطريقتين السابقتين، كما أن عملية القسمة $f(x)/f'(x)$ من الممكن أن تكون مصدر للخطورة عندما تكون $f'(x)$ صغيرة جدا أو تقترب من الصفر.

مثال ٤–٣

إفترض أننا نريد إيجاد جذر المعادلة: $f(x)=x.\sin(\pi x)-e^{-x}$. مثل هذه المعادلات يكون من الصعب إيجاد جذر لها بالطريقة التحليلية، وحتى من صيغة المعادلة يكون من الصعب توقع أين توجد أصفار مثل هذه المعادلات حتى نبحث عن القيمة الحقيقية للجذر بالقرب منها. الطريقة الفعالة فى هذه الحالة هى أن يتم رسم هذه المعادلة (ولو بطريقة خشنة) ثم نحدد بالتقريب جذور هذه المعادلة من الرسم الناتج. شكل ٤–١١ يبين رسما للمعادلة السابقة. باستخدام الأوامر التالية:

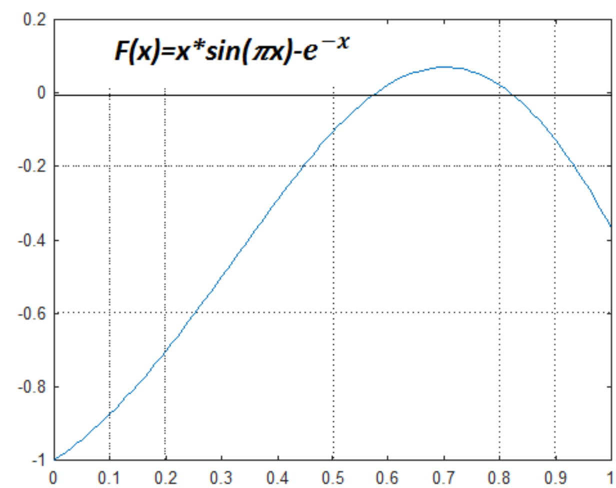
```
x=0:0.01:1;
y=x.*sin(pi*x)-exp(-x);
plot(x,y)
grid
```

كما نلاحظ من شكل ٤-١١ أن هذه المعادلة لها جذر بين x=0.5 و x=0.6 وجذر آخر بين x=0.8 و x=0.9.
سنستخدم طريقة نيوتن رافسون للبحث عن الجذر الأول وسنبدأ من نقطة البداية x0=0.1. فى هذه الحالة سيكون تفاضل المعادلة كما يلى:

$$y'=x.\pi\cos(\pi x)+\sin(\pi x)+e^{-x}$$

وسيكون خرج البرنامج كما يلى:

```
>> Newton
Enter The Function: x.*sin(pi*x)-exp(-x)
Enter Function Derivative: x*pi.*cos(pi*x)+sin(pi*x)+exp(-x)
Enter Starting Point: 0.1
Enter the Number of iterations: 100
Enter the Tolerance: 0.00001
The Root =
   0.5783
```



شكل ٤-١١ دالة المثال ٤-٣

الأن سنغير نقطة البداية x0 من x0=0.1 إلى x0=0.0 وننفذ البرنامج السابق كما يلى:

```
>> Newton
Enter The Function: x.*sin(pi*x)-exp(-x)
Ente Function Derivative: x*pi.*cos(pi*x)+sin(pi*x)+exp(-x)
Enter Startung Point: 0.0
Enter the Number of iterations: 100
Enter the Tolerance: 0.00001
The Root =
   0.8191
```

لقد استقر الخواريزم عند الجذر الثانى للمعادلة عند x=0.8191 وهذا يعكس عيبا خطيرا فى الكثير من الخواريزمات وهو حساسية الخواريزم للتغيرات البسيطة فى أحد معاملاته فكما رأينا لمجرد تغيير نقطة البداية من x=0.1 إلى x0=0.0 تغيرت نتيجة بحث الخواريزم من الجذر الأول إلى الجذر الثانى.

مثال ٤-٤

إفترض الدالة $f(x)=x^3-2x+2$. هذه الدالة لها ثلاثة جذور يمكن حسابهم مباشرة باستخدام الأمر roots() كما يلى:

```
>> a=[1 0 -2 2];
>> z=roots(a)
z =
 -1.7693 + 0.0000i
  0.8846 + 0.5897i
  0.8846 - 0.5897i
```

كما نرى هناك جذران مركبان لهذه المعادلة وجذر حقيقى عند x=-1.7693. سنحاول البحث عن هذا الجذر فى الخطوات التالية لنرى بعض المشاكل التى يمكن لبعض الخوارزمات أن تقع فيها عند ظروف معينة لبعض المعاملات.

١- إفترض نقطة البداية x0=0، بالتالى فإن f(x0)=2، و f'(x)=3x-2 و f'(x0)=-2.

٢- الآن سنحسب: x1=x0-f(x0)/f'(x0)=0-2/(-2)=1. وبالتالى فإن f(x1)=1 و f'(x1)=1.

٣- الآن سنحسب: x2=x1-f(x1)/f'(x1)=1-1/1=0. وبالتالى فإن f(x2)=2 و f'(x2)=-2 .

٤- الآن سنحسب: x3=x2-f(x2)/f'(x2)، حيث سنجد أن x3=1 ، ....

٥- وهكذا نستمر فى الحل حيث سنجد أن الخوارزم يتردد إلى مالانهاية بين قيمة xn=1 و xn+1=0 .

٦- أى أن بدأ الخوارزم من النقطة x0=0 لن يصل إلى حل. الآن دعنا نبدأ الخوارزم من x0=-3 مثلا، كما يلى:

```
>> Newton
Enter The Function: x^3-2*x+2
Enter Function Derivative: 3*x^2-2
Enter Starting Point: -3
Enter the Number of iterations: 100
Enter the Tolerance: 0.00001
The Root =
  -1.7693
```

وهو نفس الجذر الذى حصلنا عليه بالاستخدام المباشر للدالة roots() كما سبق.

٤-٤ تمارين

١- نفذ ثلاث خطوات لإيجاد جذر المعادلة: $f(x) = \sqrt{x} - \cos(x)$ فى المدى [0 1].

٢- إفترض الدالة: f(x)=3(x+1)(x-0.5)(x-1) نفذ ثلاث خطوات مستخدما القيم التالية للمدى:

(أ) [-2, 1.5]

(ب) [-1.25, 2.5]

٣- نفذ برنامج خوارزم التنصيف لإيجاد جذور المعادلة $f(x)=x^3-7x^2+14x-6$ بدقة 0.0001 فى قيم المدى التالية:

(أ) [0, 1]

(ب) [1. 3.2]

(ت) [3.2, 4]

٤- إستخدم برنامج خواريزم طريقة التنصيف لإيجاد الجذور الأربعة للمعادلة: $f(x)=x^4-2x^3-4x^2+4x+4$، بدقة 0.00001 فى الفترات التالية:

(أ) [-2, -1]

(ب) [0, 2]

(ت) [2, 3]

(ث) [-1. 0]

استخدم الدالة roots() لإيجاد جذور المعادلة مباشرة وقارن مع نتيجة البرنامج.

٥- استخدم برنامج طريقة التنصيف لحساب $\sqrt{3}$ بدقة تساوى 0.00001.

٦- استخدم برنامج طريقة التنصيف لحساب $\sqrt[3]{25}$ بدقة تساوى 0.00001.

٧- فى كل البرامج السابقة أذكر عدد الحلقات التى تمت فى البرنامج للوصول للدقة المطلوبة.

٨- نفذ ثلاث خطوات فى خواريزم الخط القاطع لإيجاد جذر المعادلة $f(x)=x^2-6$ مستخدما x0=3 و x1=2.

٩- أعد التمرين ٨ للدالة $f(x)=-x^3-\cos(x)$ مستخدما x0=-1 و x1=0.

١٠- استخدم طريقة الخط القاطع لإيجاد جذور المعادلات التالية فى المدى المرافق لكل معادلة:

(أ) الدالة: $f(x)=x^3-2x-5$ فى المدى [1, 4].

(ب) الدالة: $f(x)=x^3+3x^2-1$ فى المدى [-3, -2].

(ت) الدالة: $f(x)=x-\cos(x)$ فى المدى [0, π/2].

(ث) الدالة: $f(x)=x-0.8-0.2\sin(x)$ فى المدى [0. π/2].

١١- إستخدم طريقة الخط القاطع لإيجاد جذور المعادلات التالية بدقة تساوى 0.000001 فى المدى المبين بجوار كل منها:

(أ) الدالة: $f(x)=2x\cos(2x)-(x-2)^2$ فى المدى [2,3] والمدى [3, 4].

(ب) الدالة: $f(x)=(x-2)^2-\ln(x)$ فى الدى [1, 2].

(ت) الدالة: $f(x)=e^x-3x^2$ فى المدى [0, 1] و [3, 5].

(ث) الدالة: $f(x)=\sin(x)-e^x$ فى المدى [0, 1] والمدى [3, 4] والمدى [5, 7].

١٢- استخدم طريقة الخط القاطع لإيجاد جذر المعادلة $f(x)=4x\cos(2x)-(x-2)^2$ بدقة 0.000001 فى المدى [0. 8].

١٣- استخدم طريقة الخط القاطع لإيجاد $\sqrt{3}$ بدقة تساوى 0.00001 وقارن النتيجة مع نتيجة التمرين ٥ السابق.

١٤- استخدم طريقة الخط القاطع لإيجاد $\sqrt[3]{25}$ بدقة تساوى 10^{-6} وقارن النتيجة مع نتيجة التمرين ٦ السابق.

١٥- كثيرة الحدود التالية: $f(x)=230x^4+18x^3+9x^2-221x-9$ لها جذرين حقيقيين أحدهما فى المدى [-1,0] والثانى فى المدى [0.1]، أوجد هذين الجذرين مقربا النتيجة حتى 10^{-6} مستخدما طريقتى التنصيف والخط القاطع.

١٦- بفرض الدالة: $f(x)=x^2-6$ وأن $x0=1$ أوجد $x1$ و $x2$ و $x3$ مستخدما طريقة نيوتن.

١٧- إفترض الدالة: $f(x)=-x^3-\cos(x)$ وأن $x0=-1$. استخدم طريقة نيوتن لإيجاد $x2$. هل يمكن استخدام $x0=0$ لحل هذه التمرين.

١٨- استخدم طريقة نيوتن لإيجاد الجذور فى المدى المبين بجوار كل منها بدقة 0.00001 للدوال التالية:

(أ) الدالة: $f(x)=x^3-2x^2-5$ فى المدى [1, 4].

(ب) الدالة: $f(x)=x^3+3x^2-1$ فى المدى [-3, -2].

(ت) الدالة: $f(x)=x-\cos(x)$ فى المدى $[0, \pi/2]$.

(ث) الدالة: $f(x)=x-0.8-0.2\sin(x)$ فى المدى $[0, \pi/2]$.

١٩- استخدم طريقة نيوتن لإيجاد جذور الدوال التالية بدقة 0.000001 فى المدى المبين بجوار كل منها:

(أ) الدالة: $f(x)=2x\cos(2x)-(x-2)^2$ فى المدى [2,3] و المدى [3,4].

(ب) الدالة: $f(x)=(x-2)^2-\mathrm{lin}(x)$ فى المدى [1, 2] و المدى [e, 4].

(ت) الدالة: $f(x)=e^x-3x^2$ فى المدى [0, 1] و المدى [3,5].

(ث) الدالة: $f(x)=\sin(x)-e^{-x}$ فى المدى [0, 1] و المدى [3,4] والمدى [6, 7].

٢٠- استخدم طريقة نيوتن لإيجاد الجذور الأربعة للدالة $f(x)=4x\cos(2x)-(x-2)^2$ فى المدى [0, 8] بدقة 10^{-5}.

٢١- استخدم طريقة نيوتن لإيجاد كل جذور الدالة $f(x)=x2+10\cos(x)$ بدقة تساوى 10^{-5}.

٢٢- استخدم طريقة نيوتن لإيجاد $\sqrt{3}$ بدقة تساوى 0.00001 وقارن هذه النتيجة مع تمرينى ٥ و ١٣ السابقين.

٢٣- استخدم طريقة نيوتن لإيجاد $\sqrt[3]{\sqrt{25}}$ بدقة تساوى 10^{-6} وقارن هذه النتيجة مع تمرينى ٦ و ١٤ السابقين.

٢٤- أوجد جذر الدالة $f(x)=\tan(\pi x)-6$ فى المدى [0, 0.48] مستخدما طريقتى التنصيف والخط القاطع ثم بطريقة نيوتن مستخدما $x0=0$ مرة و $x0=0.48$ مرة أخرى. اشرح النتائج التى حصلت عليها.

٢٥- أوجد جذر الدالة: $f(x)=0.5+0.25x^2-x\sin(x)-0.5\cos(2x)$ مستخدما $x0=\pi/2$. استمر فى حلقة الحل حتى تصل إلى الدقة 10^{-5}. هل تبدو النتيجة غير طبيعية بالنسبة لطريقة نيوتن. حل المسالة أيضا مستخدما $x0=5\pi$ و $x0=10\pi$.

الفصل ٥

حل نظم المعادلات الخطية

الفصل ٥

حل نظم المعادلات الخطية

٥-١ مقدمة

الكثير من النظم الحقيقية يتم تمثيلها بعدد من المعادلات الخطية فى عدد من المجاهيل وعندما يكون عدد هذه المعادلات صغيرا، أقل من ثلاثة أو أربعة، فإنه يمكن حل هذه المعادلات يدويا أو بطرق تحليلية أو يدوية وإيجاد قيم لهذه المجاهيل بشرط أن يكون عدد المجاهيل مساويا لعدد المعادلات. أما عندما يصل عدد هذه المجاهيل وبالتالى عدد المعادلات إلى المئات أو الآلاف فإنه فى هذه الحالة يستحيل الحل التحليلى ولابد من استخدام الحاسب وبالتالى الطرق الرقمية لحل هذا النظام من المعادلات. كما ذكرنا أن كل المعادلات فى النظام لابد أن تكون خطية، وأن عدد المجاهيل يساوى عدد المعادلات. توجد نظم المعادلات الخطية فى الكثير من التطبيقات الهندسية مثل الأبنية والتركيبات، والجوامد المرنة، والتدفق الحرارى، والمجالات الكهرومغناطيسية والدوائر الكهربية وغيرها الكثير، ولذلك سنقدم فى هذا الفصل الطرق المختلفة لحل عدد n من المعادلات فى عدد n من المجاهيل. يمكن كتابة مثل هذا النظام من المعادلات الخطية كما يلى:

$$a_{11}x_1+a_{12}x_2+\dots\ \dots\dots+a_{1n}x_n=b_1$$
$$a_{21}x_1+a_{22}x_2+\dots\ \dots\dots+a_{2n}x_n=b_2$$
$$\dots\ \dots\ \dots\dots\ \dots\dots\ \dots$$
$$a_{n1}x_1+a_{n2}x_2+\dots\ \dots\dots+a_{nn}x_n=b_n$$

وهذا النظام يمكن كتابته فى صورة مصفوفات كما يلى:

$$Ax=b$$

حيث:

$$A=\begin{pmatrix} a_{11} & a_{12} \dots\dots & a_{1n} \\ a_{21} & a_{22} \dots\dots & a_{2n} \\ \dots\dots\dots & \dots\dots\dots & \dots\dots \\ a_{n1} & a_{n2} \dots\dots & a_{nn} \end{pmatrix},\qquad x=\begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}\qquad, b=\begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b \end{pmatrix}$$

والمطلوب هو حساب قيمة المتجه x الذى يحقق كل المعادلات السابقة حيث x فى كل التطبيقات تمثل استجابة أو خرج النظام و b هى الدخل للنظام و A تمثل معاملات أو خواص النظام.

هناك فئتين من فئات طرق حل نظام المعادلات الخطية. الفئة الأولى من الطرق تسمى الطرق المباشرة، والفئة الثانية تسمى الطرق التكرارية. الطرق التكرارية سيتم شرحها فى الفصل القدم، أما الفئة الأولى من الطرق والتى سنقدمها فى هذا الفصل فتعتمد على تحويل مجموعة المعادلات المعطاة إلى مجموعة أخرى من المعادلات التى يسهل حلها كما سنرى.

هذا التحويل يتم عن طريق إجراء بعض العمليات الأولية على مصفوفة المعاملات وهذه العمليات ليس لها أى تأثير على محددة هذه المصفوفة ولذلك فهى لن تؤثر على الحل النهائى وهذه العمليات هى:

١- تبديل معادلتين بين بعضهما البعض.

٢- ضرب معادلة فى ثابت لا يساوى الصفر.

٣- ضرب معادلة فى ثابت لا يساوى الصفر ثم طرحها من معادلة أخرى.

وهذه العمليات كلها سنستخدمها فى هذه الطرق المباشرة لحل نظام المعادلات.

٥-٢ طريقة جاوس لحذف المتغيرات

ربما تكون هذه الطريقة، Gauss elimination، هى الأكثر استخداما فى حل مثل هذه النظم من المعادلات الخطية. تقوم فكرة هذه الطريقة على حذف المتغيرات من المعادلات الواحد بعد الآخر حتى ننتهى بمعادلة واحدة فى متغير واحد حيث يمكن إيجاد هذا المتغير من هذه المعادلة، ثم يتم الانتقال إلى معادلة أخرى فى متغيرين أحدهما المتغير السابق الذى عرفنا قيمته ومتغير آخر يمكن حساب قيمته من هذه المعادلة الجديدة. بعد ذلك ننتقل إلى معادلة ثالثة من ثلاث متغيرات اثنان منهما المتغيران السابقان والثالث يمكن حسابه من هذه المعادلة. نستمر بهذه الطريقة حتى نصل إلى الحصول على قيم جميع المتغيرات. وأفضل طريقة لفهم ذلك تكون من خلال المثال التالى:

مثال ٥-١

افترض النظام التالى المكون من ثلاث معادلات فى ثلاث مجاهيل:

$$-3x_1+2x_2-x_3=-1$$
$$6x_1-6x_2+7x_3=-7$$
$$3x_1-4x_2+4x_3=-6$$

نبدأ هذه الطريقة بإضافة المتجه b على يمين المصفوفة A لتصبح كالتالى:

$$\begin{pmatrix} -3 & 2 & -1 & -1 \\ 6 & -6 & 7 & -7 \\ 3 & -4 & 4 & -6 \end{pmatrix}$$

تتم عملية حذف المتغيرات على صفوف هذه المصفوفة باستخدام العمليات التالية: (١) ضرب أى صف فى ثابت، (٢) إضافة العديد من الصفوف إلى صف آخر، (٣) إستبدال أى صف مكان أى صف آخر، والغرض من كل هذه العمليات هو تحويل المصفوفة الأصلية إلى مصفوفة مثلثة علوية كما سنرى.

سنبدأ باستخدام الصف الأول ونستمر فى العمل على باقى الصفوف كما يلى: أولا نريد أن نتخلص من الرقم 6 الذى فى العنصر a_{21} من الصف الثانى، لذلك سنضرب الصف الأول فى 2 ونجمعه على الصف الثانى. ولكى نتخلص من الرقم 3 الذى فى العنصر a_{31} من الصف الثالث، فإننا يكفى أن نجمع الصف الأول مع الصف الثالث ونضعهما فى الصف الثالث. نتيجة ذلك ستكون المصفوفة التالية التى تكون عناصر أول عمود فيها تساوى أصفار ماعدا أول عنصر:

$$\begin{pmatrix} -3 & 2 & -1 & -1 \\ 0 & -2 & 5 & -9 \\ 0 & -2 & 3 & -7 \end{pmatrix}$$

الآن ننتقل إلى العمود الثانى لتصفير كل العناصر التى بعد العنصر القطرى a_{22}. لعمل ذلك نضرب الصف الثانى فى -1 ونجمعه على الصف الثالث لتصبح المصفوفة الجديدة كما يلى:

$$\begin{pmatrix} -3 & 2 & -1 & -1 \\ 0 & -2 & 5 & -9 \\ 0 & 0 & -2 & 2 \end{pmatrix}$$

بذلك يصبح نظام المعادلات الأصلى كما يلى:

$$-3x_1+2x_2-x_3=-1$$
$$-2x_2+5x_3=-9$$
$$-2x_3=2$$

يمكن الآن حل هذه المعادلات عن طريق التعويض العكسى بدءا من آخر معادلة حيث سنحصل على:

$$-2x_3=2 \rightarrow x_3=-1$$
$$-2x_2=-9-5x_3=-4 \rightarrow x_2=2$$
$$-3x_1=-1-2x_2+x_3=-6 \rightarrow x_1=2$$

وعلى ذلك يمكن كتابة متجه الحل، المتجه x، كما يلى:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ -1 \end{pmatrix}$$

هذه الخطوات تسمى طريقة جاوس لحذف المتغيرات مع التعويض العكسى. يمكن تعميم هذا الحل لكى يمكن تطبيقه على أى نظام من أى عدد من المعادلات بأى عدد من المجاهيل. وسنوضح ذلك بالخطوات التالية:

سنعيد كتابة نظام المعادلات مع تسمية كل معادلة برمز E_i حيث i هى رقم المعادلة كما يلى:

$$E1:\ a_{11}x_1+a_{12}x_2+\dots\ \dots\dots+a_{1n}x_n=b_1$$
$$E2:\ a_{21}x_1+a_{22}x_2+\dots\ \dots\dots+a_{2n}x_n=b_2$$
$$Ei:\ \dots\ \dots\ \dots\dots\ \dots\dots\ \dots\dots$$
$$En:\ a_{n1}x_1+a_{n2}x_2+\dots\ \dots\dots+a_{nn}x_n=b_n$$

١- بفرض أن $a_{11}\neq 0$، نقوم بجعل كل عناصر العمود الأول بدءا من الصف الثانى (أسفل a_{11}) تساوى أصفار عن طريق العملية $(E_i-m_{i1}E_1)\rightarrow E_i$ التى تعنى ضرب كل عناصر الصف E1 فى القيمة: $m_{i1}=a_{i1}/a_{11}$، حيث i تساوى 2 و 3 و 4 حتى n. يمكن توضيح هذه العملية على المصفوفة A بعد إلحاق العمود b على يمينها كما يلى:

$$A=\begin{pmatrix} a_{11} & a_{12}\dots\dots & a_{1n} & b1 \\ a_{21} & a_{22}\dots\dots & a_{2n} & b2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ a_{n1} & a_{n2}\dots\dots & a_{nn} & bn \end{pmatrix} \quad \begin{bmatrix} E_2-m_{21}E_1 & \rightarrow & E_2 \\ E_3-m_{31}E_1 & \rightarrow & E_3 \\ \dots\dots\dots \\ E_n-m_{n1}E_1 & \rightarrow & E_n \end{bmatrix} \quad \begin{pmatrix} a_{11} & a_{12}\dots\dots & a_{1n} & b1 \\ 0 & a_{22}\dots\dots & a_{2n} & b2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ 0 & a_{n2}\dots\dots & a_{nn} & bn \end{pmatrix}$$

٢- بفرض أن العنصر القطرى $a_{22}\neq 0$، نقوم بجعل كل عناصر العمود الثانى بدءا من الصف الثالث (أسفل a_{22}) تساوى أصفار عن طريق العملية $(Ei-m_{i2}E2)\rightarrow Ei$ التى تعنى ضرب كل عناصر الصف E2 فى القيمة: $m_{i2}=a_{i2}/a_{22}$، حيث i تساوى 3 و 4 و 5 حتى n. يمكن توضيح هذه العملية على المصفوفة A بعد إلحاق العمود b على يمينها كما يلى:

$$A=\begin{pmatrix} a_{11} & a_{12}\dots\dots & a_{1n} & b1 \\ 0 & a_{22}\dots\dots & a_{2n} & b2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ 0 & a_{n2}\dots & a_{nn} & bn \end{pmatrix} \quad \begin{bmatrix} E_3-m_{32}E_2 & \rightarrow & E_3 \\ \dots\dots\dots \\ E_n-m_{n2}E_2 & \rightarrow & E_n \end{bmatrix} \quad \begin{pmatrix} a_{11} & a_{12}\dots\dots & a_{1n} & b1 \\ 0 & a_{22}\dots\dots & a_{2n} & b2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ 0 & 0\dots\dots & a_{nn} & bn \end{pmatrix}$$

بنفس الطريقة يمكن تصفير كل عناصر الأعمدة بدءا من أسفل كل عنصر قطرى إلى أن نصل إلى الصف قبل الأخير n-1. الكمية المضروبة فى صورتها العامة ستكون $m_{ki}=a_{ki}/a_{ii}$ حيث k هى رقم الصف و i هى رقم العمود، ويمكن إجراء العملية العامة: $E_k-m_{ik}E_i\rightarrow E_k$ ، حيث k تتغير من i و i+1 و i+2 و ... إلى n مع اعتبار أن العناصر القطرية a_{ii} لا تساوى صفر. بذلك نكون قد حذفنا كل العناصر تحت العنصر القطرى فى كل الأعمدة وبذلك تصبح المصفوفة كما يلى:

$$A=\begin{pmatrix} a_{11} & a_{12}\dots\dots & a_{1n} & b1 \\ 0 & a_{22}\dots\dots & a_{2n} & b2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ 0 & 0\ \dots & a_{nn} & bn \end{pmatrix}$$

لاحظ أن كل عناصر هذه المصفوفة مختلفة عن نظيرتها فى المصفوفة الأصلية فيما عدا عناصر الصف الأول فقط، ولذلك فإن تسمية هذه المصفوفة الأخيرة بنفس الإسم A تعتبر مجازية. على ضوء ذلك يمكن كتابة نظام المعادلات على الصورة التالية:

$$a_{11}x_1+a_{12}x_2+\ldots\ \ldots\ldots+a_{1n}x_n=b_1$$
$$a_{22}x_2+\ldots\ \ldots\ldots+a_{2n}x_n=b_2$$
$$\ldots\ldots\ldots\ \ \ldots\ldots$$
$$a_{nn}x_n=b_n$$

نؤكد هنا على أن عناصر نظام المعدلات السابق بما فيها العناصر b مختلفة عن نظيرتها فى النظام الأصلى (فيما عدا عناصر الصف الأول) نتيجة العمليات التى تم إجراؤها ولكننا مجازا استخدمنا نفس الرموز للتسهيل فقط. هذه الصورة لنظام المعادلات نسميها الصورة المثلثة القطرية العلوية، حيث أن كل العناصر التى على يسار القطر الأساسى تساوى أصفار. نذكر أيضا بأن كل عناصر القطر الرئيسى فى المصفوف a_{11} حتى a_{nn} يجب ألا يكون أى منها يساوى صفر، وهذا لا يعنى بالضرورة أنه لن يوجد حلا للنظام، ولكن ذلك يعنى أنه من الممكن أن يكون هناك وسائل أخرى.

من هنا نبدأ مرحلة التعويض العكسى لإيجاد قيم المجاهيل.

٣- نبدأ من آخر معادلة حيث يمكننا إيجاد x_n كما يلى: $x_n=\frac{b_n}{a_{nn}}$.

٤- ثم ننتقل إلى المعادلة قبل الأخيرة حيث يمكننا إيجاد x_{n-1} بمعلومية x_n كما يلى:

$$x_{n-1}=\frac{b_n-a_{n-1.n}x_n}{a_{n-1.n-1}}$$

٥- نستمر بنفس العملية يمكننا كتابة الصورة العامة كما يلى:

$$x_i=\frac{b_i-\sum_{j=i+1}^{n}a_{ij}x_j}{a_{ii}}$$

مثال ٥-٢

إفترض النظام الخطى التالى من أربع معادلات فى أربع مجاهيل:

$$E1:\quad x_1-x_2+2x_3-x_4=-8$$
$$E2:\quad 2x_1-2x_2+3x_3-3x_4=-20$$
$$E3:\quad x_1+x_2+x_3=-2$$
$$E4:\quad x_1-x_2+4x_3-3x_4=4$$

المصفوفة A المضاف إليها العمود b ستكون:

$$\begin{pmatrix} 1 & -1 & 2 & -1 & -8 \\ 2 & -2 & 3 & -3 & -20 \\ 1 & 1 & 1 & 0 & -2 \\ 1 & -1 & 4 & 3 & 4 \end{pmatrix}$$

لتصفير عناصر العمود الأول بدءا من الصف الثانى سنجرى العمليات التالية:

$E2-2E1 \rightarrow E2$ و $E3-E1 \rightarrow E3$ و $E4-E1 \rightarrow E4$ لنحصل على المصفوفة التالية:

$$\begin{pmatrix} 1 & -1 & 2 & -1 & -8 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & 2 & 4 & 12 \end{pmatrix}$$

كما ذكرنا من قبل عند تصفير عناصر العمود الثانى بدءا من الصف الثالث، أى العناصر التى أسفل العنصر القطرى a_{22}، فإننا سنحتاج لمعامل الضرب $m_{ki}=a_{ki}/a_{ii}$ حيث a_{ii} فى هذه الحالة هو العنصر a_{22} فى المصفوفة السابقة والذى يساوى صفر. أى أننا سنقسم على صفر!! فما العمل؟ الحل فى هذه الحالة هو استبدال أى صف من الصفوف التالية للصف الثانى مع الصف الثانى وهذا مسموح به حيث لن يؤثر على نظام المعادلات. لذلك فإن العملية التالية ستكون $E2 \leftrightarrow E3$. وبالتالى ستصبح المصفوفة الجديدة هى:

$$\begin{pmatrix} 1 & -1 & 2 & -1 & -8 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 2 & 4 & 12 \end{pmatrix}$$

لاحظ أن العنصر الأخير فى العمود الثانى قد تم تصفيره بالصدفه. الآن ننتقل العمود الثالث حيث نريد تصفير العنصر الأخير فى هذا العمود فقط. يمكن إجراء ذلك عن طريق العملية $E4+2E3 \rightarrow E4$ وهذا يعطينا المصفوفة التالية:

$$\begin{pmatrix} 1 & -1 & 2 & -1 & -8 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 0 & 2 & 4 \end{pmatrix}$$

بذلك نكون قد انتهينا من تحويل المصفوفة A الأصلية إلى مصفوفة مثلثة علوية قطرية ويمكن كتابة نظام المعادلات الجديد كما يلى:

$$\begin{aligned} E1:&\quad x_1 - x_2 + 2x_3 - x_4 = -8 \\ E2:&\quad + 2x_2 - x_3 + x_4 = 6 \\ E3:&\quad - x_3 - x_4 = -4 \\ E4:&\quad 2x_4 = 4 \end{aligned}$$

من المعادلة E4 نجد أن $x_4=4/2=2$.

بالتعويض بقيمة x_4 فى المعادلة E3 نحصل على: $x_3=(-4+2)(-1)=2$.

بالتعويض بقيمة x_4 و x_3 فى المعادلة E2 نحصل على: $x_2=(6-2+2)2=3$.

بالتعويض بقيمة x_4 و x_3 و x_2 فى المعادلة E1 نحصل على: $x_1=(-8+2-2x2+3)=-7$.

وعلى ذلك فإن متجه الحل x يمكن كتابته كما يلى:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} -7 \\ 3 \\ 2 \\ 2 \end{pmatrix}$$

٥-٣ طريقة جاوس لحذف المتغيرات مع المحورة

لقد رأينا فى مثال ٥-٢ كيف أنه فى أى مرحلة من مراحل الحل عندما يكون العنصر القطرى a_{ii} يساوى الصفر فإنه حتى لا نتعرض لأخطار القسمة على الصفر فإننا نقوم باستبدال هذا الصف الذى يحتوى a_{ii} مع أى صف آخر تالى له باستخدام العملية Ei↔Ej التى لا تؤثر على المصفوفة أو على نتيجة الحل. سنرى فى هذا الجزء أنه لتقليل خطأ التقريب نتيجة عدد الخانات المحدد فى العمليات الحسابية فإنه من الضرورى أن نقوم بإجراء عملية الاستبدال على الصفوف حتى لو لم يكن العنصر a_{ii} يساوى صفرا. هذه العملية تسمى المحورة pivoting.

لقد رأينا أننا فى كل مرحلة نستخدم عملية القسمة $m_{ki} = {}^{a_{ki}}/_{a_{ii}}$ فى عمليات تصفير العناصر التى تقع أسفل العنصر المحورى أو القطرى a_{ii}. فى هذه القسمة إذا كان العنصر a_{ii} صغيرا جدا فإن مقدار الكمية m_{ki} سيكون كبيرا جدا مما سيتسبب فى خطأ تقريب عند استخدام هذه الكمية الكبيرة فى عمليات حسابية أخرى مثل الضرب أو الطرح من عناصر أخرى مما قد يتسبب فى خطأ الحذف الذى تكلمنا عنه من قبل. فمثلا فى مراحل التعويض العكسى فإننا نحسب قيمة x_i باستخدام المعادلة:

$$x_i = \frac{b_i - \sum_{j=i+1}^{n} a_{ij}}{a_{ii}}$$

عندما يكون العنصر a_{ii} صغيرا جدا فإن أى خطأ تقريب أو حذف نتيجة عملية الطرح التى فى البسط سيتم تكبيره بدرجة كبيرة جدا نتيجة القسمة على هذه الكمية الصغيرة a_{ii}. المثال التالى سيوضح ذلك:

مثال ٥-٣

إفترض نظام المعادلات الخطى التالى فى المتغيرين x_1 و x_2:

$$E1:\quad 0.003000x_1 + 59.14x_2=59.17$$
$$E2:\quad 5.291x_1 - 6.130x_2=46.78$$

الحل الصحيح لهذا النظام هو:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 10.00 \\ 1.000 \end{pmatrix}$$

سنفترض الآن أننا سنحل هذا النظام باستخدام طريقة جاوس للحذف مستخدمين دقة حساب من أربع خانات عشرية. لكى نصفر العنصر الأول من الصف الثانى فإننا نستخدم العملية: $E2-m_{21}E1 \rightarrow E2$ حيث:

$$m_{21} = \frac{5.291}{0.003000} = 1763.6$$

وباستخدام التقريب لأربع خانات عشرية يصبح هذا الثابت: $m_{21}=1764$. ويصبح نظام المعادلات الجديد كالتالى:

$$0.003000x_1+59.14x_2=59.17$$
$$-104300x_2=-104400$$

ومنها نحصل على قيم المتغيرات x_1 و x_2 التالية:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -10.00 \\ 1.001 \end{pmatrix}$$

حيث يوجد خطأ صغير فى قيمة x_2 وخطأ كبيرا جدا فى قيمة x_1 التى أصبحت -10 بدلا من 10. لقد تم حساب قيمة x_1 من المعادلة التالية:

$$x_1 \approx \frac{59.17-(59.14)(1.001)}{0.003000} = -10.00$$

حيث ما حدث هنا هو أن الخطأ 0.001 فى قيمة x_2 فى البسط تم ضربه فى القيمة $59.14/0.003000 \approx 20000$ مما تسبب فى هذا الخطأ الكبير فى قيمة x_1. فى عدم وجود هذا الخطأ ستكون:

$$x_1 \approx \frac{59.17-(59.14)(1)}{0.003000} = 10.00$$

لقد أوضح هذا المثال خطورة أن يكون العنصر المحورى أو القطرى فى أى مرحلة من مراحل الحل باستخدام طريقة جاوس لحذف المتغيرات صغيرا جدا حيث فى النظم الكبيرة من الممكن أن تكون نتيجة مثل هذا الخطأ أخطر من ذلك بكثير. لكى نتجنب مثل هذا الخطأ فإننا عند الخطوة i فى الحل بطريقة جاوس سننظر فى كل عناصر العمود i بدءا من العنصر القطرى a_{ii} حتى آخر عنصر فى هذا العمود بحثا عن أكبر عنصر a_{ij} (حيث j تتغير من i حتى آخر عنصر n فى هذا العمود) ونستبدل الصف الذى يقع به هذا العنصر مع الصف الذى به العنصر القطرى a_{ii}. ويمكن التعبير عن ذلك كما يلى:

$$|a_{pi}| = \max_{i \le k \le n} |a_{ki}|$$

حيث بعدها نقوم بعملية الاستبدال التالية $Ei \leftrightarrow Ep$. لاحظ أنه فى هذه الحالة يتم الاستبدال على الصفوف فقط ولا يتم استبدال على أى أعمدة. سنرى نتيجة تطبيق عملية الاستبدال هذه أو عملية المحورة على المثال السابق كما يلى:

$$E1:\quad 0.003000x_1 + 59.14x_2=59.17$$
$$E2:\quad 5.291x_1 - 6.130x_2=46.78$$

فى هذه الحالة سيكون العنصر $a_{21}=5.291$ أكبر من العنصر a_{11}، زبالتالى سنقوم بعملية الاستبدال كما يلى:

$$E1:\quad 5.291x_1 - 6.130x_2=46.78$$
$$E2:\quad 0.003000x_1 + 59.14x_2=59.17$$

فى هذه الحالة لكى نصفر العنصر a_{21} سنجرى العملية $E2\text{-}m_{21}E1 \rightarrow E2$ حيث:

$$m_{21}=(a_{21}/a_{11})=0.003000/5.291=0.0005670$$

وسيصبح نظام المعادلات بعد هذه العملية كما يلى:

$$E1:\quad 5.291x_1 - 6.130x_2=46.78$$
$$E2:\quad 59.14x_2 \approx 59.17$$

وبالتالى بالتعويض العكسى سنحصل على القيم الصحيحة لكل من x_1 و x_2 . هذه الطريقة تسمى طريقة جاوس لحذف المتغيرات مع المحورة الجزئية، أو اختصارا المحورة الجزئية partial pivoting.

٥-٤ خواريزم جاوس مع المحورة الجزئية

سنفترض أن لدينا نظام عام من المعادلات يتكون من عدد n من المعادلات فى n من المجاهيل وسنعيد كتابته هنا للتذكرة فقط:

$$E1:\quad a_{11}x_1+a_{12}x_2+\ldots\ \ldots\ldots+ a_{1n}x_n=b_1$$
$$E2:\quad a_{21}x_1+a_{22}x_2+\ldots\ \ldots\ldots+ a_{2n}x_n=b_2$$
$$Ei:\quad \ldots\ \ldots\ \ldots\ \ldots\ \ldots$$
$$En:\quad a_{n1}x_1+a_{n2}x_2+\ldots\ \ldots\ldots+ a_{nn}x_n=b_n$$

والذى يمكن كتابته فى صورة المصفوفة كما ذكرنا $Ax=b$، حيث A هى مصفوفة معاملات النظام وهى مصفوفة مربعة nxn و x هى متجه المجاهيل و b هى مصفوفة الثوابت. سنقدم هنا خطوات إجراء مثل هذا الخواريزم بحيث يمكن للقارىء أن يترجم هذه الخطوات بأى لغة برمجة يريدها مثل الماتلاب.

١- الدخل لهذا الخواريزم سيكون مصفوفة المعاملات A و n التى تمثل بعد هذه المصفوفة، ومصفوفة الثوابت b.

٢- نتيجة هذا الخواريزم ستكون المتجه x الذى يمثل متجه الحلول من عدد من العناصر، أو رسالة خطأ err تبين أن هذا الخواريزم لا يمكن تنفيذه بسبب خطأ معين ويتم بيان هذا الخطأ.

٣- **الحلقة الأولى**: متغيرها هو k التى تتغير من واحد إلى n وهى حلقة على الأعمدة تبدأ من العمود الأول ثم الثانى وهكذا حتى العمود n-1 (قبل الأخير).

٤- سنبحث فى هذا العمود k بدءا من الصف رقم k حتى الصف رقم n (الأخير) عن العنصر ذو القيمة العظمى وسنحدد رقم هذا الصف im وقيمته العظمى amax كالتالى:

$amax = \max_{k\le i\le n} |a_{ik}|$ و im هى رقم الصف الذى سيحتوى القيمة amax.

٥- إذا كانت $|amax|=0$ فإن ذلك يعنى أنه قد تم بحث كل عناصر العمود k بدءا من الصف رقم k ووجدنا أن كل هذه العناصر تساوى أصفار، وهذا يعنى أن هذه المصفوفة A أحادية singular وبالتالى فلا يوجد حل ويجب إعطاء رسالة خطأ err تبين أن المصفوفة أحادية ويتم الخروج من البرنامج. يجب أن نتذكر أنه من شرط أن يكون لهذا النظام حلا أن تكون المصفوفة A غير أحادية أو أن محددها لا يساوى الصفر.

٦- إذا كانت im=k فى الخطوة رقم ٥ ننتقل إلى الخطوة رقم ٩ مباشرة. معنى أن im=k أنه بالبحث وجد أن الصف رقم k (الذى بدأت عنده عملية البحث) هو الذى يحتوى العنصر ذو القيمة العظمى ولذلك فإنه لا حاجة لاستبدال الصفوف. إذا كانت $im\neq k$ ننتقل إلى الخطوة ٧ للبدأ فى عملية استبدال الصفين.

٧- هنا (عندما $im\neq k$) سنستبدل كل عناصر الصف رقم im مع عناصر الصف رقم k كما يلى: $a_{k,j}\leftrightarrow a_{im,j}$ لكل قيم j من k حتى n.

٨- لا تنسى أن تجرى الاستبدال على القيمة الثابتة b فى نفس الصف أيضا كالتالى: $b_k\leftrightarrow b_{im}$. فى بعض الأحيان يفضل البعض إلحاق عمود الثوابت b كصف إضافى على يمين المصفوفة A. فى هذه الحالة لن نحتاج إلى الخطوة ٨ وستكون الخطوة ٧ كما هى سوى أن j ستتغير من k حتى n+1.

حتى هنا تكون عملية استبدال الصف الذى يحتوى القيمة العظمى فى العمود k مع الصف رقم k قد تمت وهى ما يسمى بعملية المحورة، حيث فى هذه الحالة نكون قد تأكدنا أن العنصر a_{kk} له أكبر قيمة مطلقة فى العمود k. والآن سنبدأ فى عملية تصفير عناصر الصف k بدءا من الصف رقم k+1، عن طريق الضرب فى معامل الضرب والطرح كما فى الخطوة التالية مع ملاحظة أن الحلقة الأولى لم تنتهى بعد وما زلنا فى داخلها:

٩- **الحلقة الثانية**: سنبدأ من هنا حلقة ثانية داخل الحلقة الأولى لإجراء عملية التصفير على كل عناصر العمود k بدءا من الصف k+1 وحتى الصف n، ومتغير هذه الحلقة هو i حيث i تتغير من الصف k+1 حتى الصف n.

١٠ – **الحلقة الثالثة**: هذه الحلقة فى داخل الحلقة الثانية. فى هذه الحلقة سنحسب $a_{ij} = a_{ij} - ma_{kj}$ حيث m هى معامل الضرب التالى: $m=a_{ik}/a_{kk}$، حيث j تتغير من العمود k حتى العمود n. لاحظ أن m تكون ثابتة على الصف الواحد. ولا تنسى أن تحسب أيضا $b_i=b_i-mb_k$.

١١ – نهاية الحلقة الثالثة.

١٢ – نهاية الحلقة الثانية.

١٣ – نهاية الحلقة الأولى.

١٤ – إلى هنا تنتهى مرحلة وضع المصفوفة A فى الصورة المثلثة العلوية ويمكننا البدء فى مرحلة التعويض العكسى لإيجاد قيم المجاهيل بدءا من آخر مجهول xn.

١٥ – كما رأينا ستكون قيمة x_n تساوى: $x_n=b_n/a_{nn}$.

١٦ – الحلقة الرابعة: هذه الحلقة سيكون متغيرها هو i حيث i تتغير من الصف n-1 حتى الصف الأول وفى كل مرة نحسب قيمة x_i كما يلى:

$$x_i = (b_i - \sum_{j=i+1}^{n} a_{ij}x_j)/a_{ii}$$

١٧ – نهاية الحلقة الرابعة.

١٨ – خروج من البرنامج وطباعة المتجه x.

هناك دالة جاهزة فى ماتلاب يمكن استخدامها ل حساب amax مباشرة وهى:

[amax, im]=max(abs(A(k:n,k)))

حيث تبحث هذه الدالة فى العمود k من المصفوفة A بدءا من الصف k حتى الصف n عن أكبر قيمة مطلقة ويضعها فى amax ويضع رقم الصف الذى وجدت فيه هذه القيمة العظمى فى im.

البرنامج التالى عبارة عن مقترح لتنفيذ خواريزم جاوس لحذف المتغيرات مع المحورة. بالطبع يمكن للقارىء أن يضع مقترحة سواء بلغة الماتلاب أو أى لغة أخرى.

```
%program to solve a system of linear equations
clear ;
format short
a = input([' Enter The Array of Coefficients a= ']); %e.g [1 2;3 4]
b = input([' Enter The Arrays of Constants b = ' ]) ;
n = size(a,1) ;
A = [ a , b ] ; % indent a and b in a single matrix
for k=1:n %loop on the columns
[amax, im]=max(abs(A(k:n,k)));
      if amax==0
         disp('matrix is singular, no possible solution');
         break;
```

```
        end
        im=im+k-1;
        if im~=k
            T = A(k,:);
            A(k,:) = A(im,:);
            A(im,:) = T;
        end
        for i=k+1:n
          A(i,k:n+1)=A(i,k:n+1)-(A(i,k)/A(k,k))*A(k,k:n+1);
        end
        end % end of triangularization of the matrix
        if A(n,n)==0
            disp('matrix is singular, no possible solution ann=0');
            break;
        end
        x(n)=A(n,n+1)/A(n,n); %start of backward substitution
        for i=n-1:-1:1
           x(i)=(A(i,n+1)-sum(A(i,i+1:n).*x(i+1:n)))/A(i,i);
        end
        disp(x)
```

بتنفيذ بعض الأمثلة السابقة على نفس البرنامج للتحقق من صحته نحصل على ما يلى:

بالنسبة للمثال ٥-٣:

$$E1:\quad 0.003000x_1 + 59.14x_2 = 59.17$$
$$E2:\quad 5.291x_1 - 6.130x_2 = 46.78$$

>> GElimination

Enter The Array of Coefficients a= [0.003 59.14;5.291 -6.130]

Enter The Arrays of Constants b = [59.17;46.78]

x =

   10   1

وهى نفس الإجابة التى حصلنا عليها بالحل اليدوى.

بالنسبة للمثال ٥-٣:

$$E1:\quad x_1 - x_2 + 2x_3 - x_4 = -8$$
$$E2:\quad 2x_1 - 2x_2 + 3x_3 - 3x_4 = -20$$
$$E3:\quad x_1 + x_2 + x_3 = -2$$
$$E4:\quad x_1 - x_2 + 4x_3 - 3x_4 = 4$$

>> GElimination

Enter The Array Coefficients a= [1 -1 2 -1 ;2 -2 3 -3 ; 1 1 1 0 ;1 -1 4 3]

Enter The Arrays Constants b = [-8;-20;-2;4]

x =

-7.0000 3.0000 2.0000 2.0000

وهى نفس الإجابة التى حصلنا عليها بالحل اليدوى أيضا. ويمكن للقارىء التجربة مع أمثلة أخرى أو نظم المعادلات التى سنقدمها فى تمارين نهاية الفصل.

٥-٥ تحليل المصفوفات إلى مصفوفتين علوية وسفلية

إذا أمكن تحليل مصفوفة المعاملات A إلى مصفوفتين علوية upper، حيث المصفوفة العلوية هى المصفوفة التى تكون عناصرها التى تحت القطر الرئيسى تساوى أصفار، وأخرى سفلية lower، حيث المصفوفة السفلية هى المصفوفة التى كل عناصرها فوق المحور الرئيسى تساوى أصفار مع كون كل عناصر القطر الرئيسى تساوى وحايد، فإنه يمكن استخدام ذلك فى حل نظام المعادلات الخطية بطريقة أكثر كفاءة كما سنرى. لتوضيح ذلك سنقدم المثال التالى:

مثال ٥-٤

إفترض النظام الخطى التالى:

$$x_1+x_2 \quad +3x_4=4$$
$$2x_1+x_2-x_3+x_4=1$$
$$3x_1-x_2-x_3+2x_4=-3$$
$$-x_1+2x_2+3x_3-x_4=4$$

يمكن باستخدام طرق الجبر الخطى التى من المفروض أن يكون سبق دراستها أو يمكن مراجعتها فى أى كتاب عن الجبر الخطى (أنظر الفصل ٢) أن نضع مصفوفة المعاملات A لنظام المعادلات السابق فى صورة مصفوفتين علوية وسفلية كما يلى:

$$A=\begin{pmatrix}1&1&0&3\\2&1&-1&1\\3&-1&-1&2\\-1&2&3&-1\end{pmatrix}=\begin{pmatrix}1&0&0&0\\2&1&0&0\\3&4&1&0\\-1&-3&0&1\end{pmatrix}\begin{pmatrix}1&1&0&3\\0&-1&-1&-5\\0&0&3&13\\0&0&0&-13\end{pmatrix}=LU$$

هذا التحليل لمصفوفة المعاملات يساعد فى حل نظام المعادلات السابق كما يلى:

$$\begin{pmatrix}1&1&0&3\\2&1&-1&1\\3&-1&-1&2\\-1&2&3&-1\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\end{pmatrix}=\begin{pmatrix}8\\7\\14\\-7\end{pmatrix}$$

والذى يمكن كتابته على الصورة LUx=b كما يلى:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ -1 & -3 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 3 \\ 0 & -1 & -1 & -5 \\ 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & -13 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 8 \\ 7 \\ 14 \\ -7 \end{pmatrix}$$

فى المعادلة السابقة سنضع y=Ux وبالتالى فإن Ly=b ويمكن كتابة المعادلة السابقة كما يلى:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ -1 & -3 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 8 \\ 7 \\ 14 \\ -7 \end{pmatrix}$$

بالتعويض الأمامى فى المعادلة السابقة يمكن إيجاد المتجه y كما يلى:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 8 \\ -9 \\ 26 \\ -26 \end{pmatrix}$$

بذلك يمكن استخدام المتجه y فى حل المعادلة Ux=y كما يلى:

$$\begin{pmatrix} 1 & 1 & 0 & 3 \\ 0 & -1 & -1 & -5 \\ 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & -13 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 8 \\ -9 \\ 26 \\ -26 \end{pmatrix}$$

حيث بالتعويض العكسى يمكن حساب المتجه x:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 3 \\ -1 \\ 0 \\ 2 \end{pmatrix}$$

إن الحل بهذه الطريقة قد يبدو مطولا بعض الشىء عند مقارنته بطريقة الحل باستخدام طريقة جاوس لحذف المتغيرات مع المحورة، ولكنه به خاصية مهمة تفضل استخدامه فى بعض المواقف. فى الكثير من التطبيقات العملية يكون نظام المعادلات Ax=b به مصفوفة المعاملات ثابتة ولا تتغير ولكن متجه الثوابت b يتغير بتغير ظروف التطبيق، والمطلوب هو إيجاد المتجه x مع كل قيمة مناظرة للمتجه b. فى هذه الحالة لو استخدمنا طريقة جاوس فإنه لابد من إجراء عمليات جاوس للحذف مع كل قيمة للمتجه b كما لو كانت مسألة جديدة تماما. ولكن مع استخدام طريقة التحليل العلوى السفلى فإنه سيتم تحليل المصفوفة A مرة واحدة فقط، وهذا التحليل نفسه يتم استخدامه مع كل قيمة للمتجه b دون الحاجة لإجراء التحليل من جديد وذلك لأن التحليل لا يتعامل مع المتجه b ولكن طريقة جاوس كانت تتعامل مع المتجه b أثناء عمليات الحذف للمتغرات واستبدال المعادلات.

هناك ميزة أخرى فى استخدام الصورة المحللة لمصفوفة المعاملات حيث أنه من المعروف أن عدد العمليات (ضرب وقسمة وجمع وطرح) اللازمة لإجراء طريقة جاوس تتناسب مع n^3 حيث n هى عدد عناصر المصفوفة ويرمز لذلك فى المراجع بالرمز $O(n^3)$. عندما تكون المصفوفة فى صورة مثلثة علوية فإن حل النظام سيتطلب عدد من العمليات فى هذه الحالة

يتناسب مع n^2، أى $O(n^2)$. كما رأينا فى المثال السابق عند تحليل مصفوفة المعاملات إلى مصفوفة مثلثة علوية وأخرى سفلية فإن الحل سيكون من خطوتين كل منهما سيتناسب فيه عدد العمليات مع n^2 أى أن العدد الكلى للعمليات سيكون أيضا $O(n^2)$. أى أن عدد العمليات المطلوبة لحل النظام Ax=b سيقل من $O(n^3)$ فى حالة جاوس إلى $O(n^2)$ فى حالة استخدام الصورة المحللة لمصفوفة المعاملات إلى الصورة العلوية السفلية، وهذه ميزة كبيرة جدا بالذات عندما تكون n كبيرة. فمثلا عندما تكون n=100 فإن 100^2=10000 بينما 100^3=1000000 أى أن هناك تخفيض يصل إلى ٩٩% فى العمليات المطلوبة. ربما يقول البعض أن عملية تحليل المصفوفة إلى الصورة العلوية السفلية LU تحتاج لعمليات أيضا، وهذا صحيح فهى تحتاج لعمليات تتناسب مع $O(n^3)$، ولكن العزاء هو عند استخدامها مع النظم التى لها العديد من مصفوفة الثوابت b حيث فى هذه الحالة سيتم التحليل لمرة واحدة فقط.

على الرغم من ذلك فإنه يمكن استخدام طريقة جاوس بطريقة أكثر سرعة فى حالة وجود أكثر من متجه للثوابت b ولكنها بالطبع لن تكون أسرع من طريقة تحليل مصفوفة المعاملات إلى مصفوفة علوية سفلية. تعتمد هذه الطريقة على إلحاق كل متجهات الثوابت على يمين مصفوفة المعاملات ثم يتم إجراء مرحلة حذف المتغيرات باستخدام طريقة جاوس. بعد ذلك تبدأ مرحلة التعويض العكسى باستخدام كل متجه من متجهات الثوابت على حده، وسنوضح ذلك بالمثال التالى:

مثال ٥-٥

استخدم طريقة جاوس مع المحورة لحل نظام المعادلات Ax=b حيث:

$$b2=\begin{pmatrix}22\\-18\\7\end{pmatrix} \quad و \quad b1=\begin{pmatrix}-14\\36\\6\end{pmatrix} \quad و \quad A=\begin{pmatrix}6&-4&1\\-4&6&-4\\1&-4&6\end{pmatrix}$$

سيكون الحل بنفس طريقة جاوس السابقة حيث سنقوم بإلحاق المتجهين b1 و b2 على يمين المصفوفة A كما يلى:

$$\begin{pmatrix}6&-4&1&-14&22\\-4&6&-4&36&-18\\1&-4&6&6&7\end{pmatrix}$$

أول خطوة من مرحلة الحذف أو تصفير معاملات العمود الأول ستكون بضرب الصف الأول فى (2/3) والجمع مع الصف الثانى ونضع فى الصف الثانى، وبضرب الصف الأول فى (1/6-) والجمع مع الصف الثالث ونضع فى الصف الثالث لنحصل على المصفوفة التالية:

$$\begin{pmatrix}6&-4&1&-14&22\\0&10/3&-10/3&80/3&-10/3\\0&-10/3&35/6&25/3&10/3\end{pmatrix}$$

الخطوة الثانية سنجمع الصف الثانى مع الصف الثالث ونضع فى الصف الثالث لنحصل على المصفوفة التالية:

$$\begin{pmatrix} 6 & -4 & 1 & -14 & 22 \\ 0 & 10/3 & -10/3 & 80/3 & -10/3 \\ 0 & 0 & 5/2 & 35 & 0 \end{pmatrix}$$

فى مرحلة التعويض العكسى سنستخدم المتجه الموجود فى العمود الرابع فى المصفوفة السابقة لحساب متجه الحل المقابل له x1:

$$x1 = \begin{pmatrix} 10 \\ 22 \\ 14 \end{pmatrix}$$

ونستخدم العمود الخامس فى المصفوفة السابقة لحساب متجه الحل الثانى x2 كما يلى:

$$x2 = \begin{pmatrix} 3 \\ -1 \\ 0 \end{pmatrix}$$

٥-٦ حل نظام المعادلات الخطية باستخدام ماتلاب مباشرة

يوفر الماتلاب طرق مباشرة لحل أى نظام خطى من المعادلات وأول هذه الطرق هى الدالة linsolve(A,b) حيث A مصفوفة المعاملات و b هى متجه الثوابت، ويمكن التحقق من ذلك بالتطبيق على الأمثلة السابقة فى ماتلاب مباشرة كما يلى:

```
>> A=[1 -1 2 -1 ;2 -2 3 -3 ; 1 1 1 0 ;1 -1 4 3 ];
>> b=[-8;-20;-2;4];
>> X = linsolve(A,b)
X =
  -7.0000
   3.0000
   2.0000
   2.0000
```

الدالة ()linsolve تستخدم تحليل مصفوفة المعاملات إلى مصفوفة مثلثة علوية وأخرى سفلية مع المحورة الجزأية للحصول على النتيجة. وطالما أنه سيستخدم المحورة الجزأية فإن النتيجة ستكون فى الغالب دقيقة جدا لأن المحورة تتجنب أخطاء التقريب كما رأينا. أنظر إلى المثال التالى أيضا:

```
>> A=[0.003 59.14;5.291 -6.130];
>> b=[59.17;46.78];
>> X=linsolve(A,b)
X =
   10
```

```
    1
```

يمكن استخدام ماتلاب لحساب حل نظام المعادلات الخطية باستخدام معكوس مصفوفة المعاملات. نحن نعلم من الجبر الخطى أن معكوس أى مصفوفة هى مصفوفة لو تم ضربها فى المصفوفة الأصلية كانت النتيجة هى مصفوفة الوحدة. أى أن $AA^{-1}=I$ حيث A^{-1} هى معكوس المصفوفة A و I هى مصفوفة الوحدة. بكتابة نظام المعادلات الخطية فى صورة المصفوفية $Ax=b$ فإنه بضرب الطرفين فى معكوس المصفوفة A نحصل على $x=A^{-1}b$. أى أنه يمكن الحصول على متجه المجاهيل x بضرب متجه الثوابت b من اليسار فى معكوس مصفوفة المعاملات A. يمكن التعبير عن ذلك فى ماتلاب كما يلى:

```
>> A=[0.003 59.14;5.291 -6.130];
>> b=[59.17;46.78];
>> X=inv(A)*b
X =
  10.0000
   1.0000
```

ويمكن استخدام عملية القسمة العكسية x=A\b فى ماتلاب كما يلى:

```
>> A=[0.003 59.14;5.291 -6.130];
>> b=[59.17;46.78];
>> x=A\b
x =
   10
    1
```

مثال ٥-٦

استخدم طريقة جاوس والطرق المباشرة فى ماتلاب للحصول على حل لمجموعة المعادلات الخطية التالية:

$$10x_1 \quad -7x_2 \quad =7$$
$$-3x_1+2.099x_2+6x_3=3.901$$
$$5x_1 \quad -x_2+5x_3=6$$

الحل اليدوى الصحيح لهذا النظام هو:

$$X=\begin{pmatrix}0\\-1\\1\end{pmatrix}$$

باستخدام طريقة جاوس مع المحورة نحصل على ما يلى بالتطبيق المباشر فى الخواريزم السابق:

```
>> GElimination
 Enter The Array of Coefficients a= [10 -7 0;-3  2.099  6; 5 -1 5]
 Enter The Arrays of Constants b = [7;3.901;6]
   0.0000   -1.0000    1.0000
```

وهى نتيجة صحيحة تماما.

باستخدام طريقة معكوس المصفوفة مباشرة فى ماتلاب نحصل على:

```
>> A=[10 -7 0;-3  2.099 6; 5 -1 5];
>> b=[7;3.901;6];
>> X=A\b
X =
    0
   -1
    1
```

وهى نتيجة صحيحة تماما أيضا.

باستخدام الدالة linsolve():

```
>> A=[10 -7 0;-3  2.099 6; 5 -1 5];
>> b=[7;3.901;6];
>> linsolve(A,b)
ans =
    0
   -1
    1
```

وهى نفس النتيجة أيضا. إستخدم أى طريقة تفضلها.

٥-٧ أصالة أو تفرد الحل

كلمة أخيرة نضيفها هنا قبل الانتهاء من هذا الفصل وهى عن شروط الحصول على حل، وحل فريد لمجموعة المعادلات الخطية. لقد ذكرنا من قبل أنه لكى يكون هناك حل لهذا النظام من المعادلات الخطية فإن مصفوفة المعاملات يجب ألا تكون متفردة singular أو استثنائية، (ونحن نعرف من الجبر الخطى أن المصفوفة المتفردة يكون محددها يساوى صفر، بمعنى $|A|=0$) وإذاكانت مصفوفة المعاملات متفردة فإنه إما أنه لن يكون هناك حل على الإطلاق، أو سيكون هناك عدد لانهائى من الحلول اعتمادا على متجه الثوابت. إفترض مثلا المعادلة:

$$2x_1+x_2=3$$

والمعادلة:

$$4x_1+2x_2=6$$

توجد علاقة بين المعادلتين حيث أن المعادلة الثانية يمكن الحصول عليها من المعادلة الأولى بضربها فى اثنين، وبالتالى فإن أى قيمتين لكل من x_1 و x_2 تحقق المعادلة الأولى ستعتبر حلا للمعادلة الثانية، وبالطبع هناك مالا نهاية من هذه الحلول.

أنظر أيضا للمعادلتين التاليتين:

$$2x_1+x_2=3$$

والمعادلة:

$$4x_1+2x_2=0$$

بقسمة المعادلة الثانية على اثنين نحصل على نفس الطرف الأيسر من المعادلة الأولى، مما يعنى أن الطرف الأيسر فى المعادلتين سيكون نفسه والطرف الأيمن مختلف، وهذا يعتبر تعارض سيتسبب فى عدم وجود حل على الإطلاق للمعادلتين.

يأتى هنا سؤال مشروع وهو ماذا لو كانت المصفوفة قريبة من التفرد، أى أن محددها لا يساوى الصفر تماما ولكنه قريب من الصفر جدا، أو بمعنى آخر صغيرا جدا؟ لذلك فنحن بحاجة هنا إلى مقياس نقيس به محدد المصفوفة لنحكم عليه إذا كان صغيرا أم لا. هذا المقياس أو المرجع يسمى معيار المصفوفة أو matrix norm ورمزه فى معظم المراجع هو $\|A\|$. وفى هذه الحالة نقول أن محدد المصفوفة صغير إذا كان:

$$|A| \ll \|A\|$$

هناك العديد من المعايير المعرفة فى المطبوعات المتخصصة والتى منها التعريف التالى:

$$\|A\| = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}^2}$$

والذى يساوى الجذر التربيعى لمجموع مربعات جميع عناصر المصفوفة. هناك مقياس رسمى لمدى تكيف المصفوفة وهو رقم تكيف المصفوفة matrix condition number ويعطى بالمعادلة التالية:

$$cond(A) = \|A\|\|A^{-1}\|$$

إذا كان هذا الرقم قريبا من الواحد فإن المصفوفة تكون جيدة التكيف، ويزداد مع زيادة درجة سوء التكيف، حتى يصل إلى مالانهاية عندما تكون المصفوفة متفردة. لاحظ أن هذا الرقم لا يكون وحيدا ولكنه سيعتمد على طريقة اختيار المعيار norm. لسوء الحظ فإن حساب رقم التكيف يأخذ الكثير من الحساب والوقت وبالذات للمصفوفات الكبيرة. لذلك فى معظم الأحيان يكفى لقياس تكيف المصفوفة أن يتم مقارنة محددها مع مقادير عناصرها.

عندما تكون مصفوفة نظام المعادلات سيئة التكيف يكون حل هذا النظام حساسا جدا للتغيرات الصغيرة جدا فى عناصر مصفوفة المعاملات، ونبين ذلك بفرض نظام المعادلات التالى:

$$2x_1+x_2=3$$
$$2x_1+1.001x_2=0$$

هذا النظام له الحل التالى: $x_1=1501.5$ و $x_2=-3000$. لاحظ أن محدد المصفوفة $|A|=2(1.001)-2(1)=0.002$ وهو أصغر كثيرا جدا من عناصر المصفوفة، ولذلك فهذا النظام للمعادلات سيء التكيف. إذا قمنا بتغيير أحد عناصر المصفوفة تغيرا بسيطا بأن نجعل المعادلة الثانية كالتالى: $2x_1+1.002x_2=0$ فإن حلول النظام ستصبح $x_1=751.5$ و $x_2=-1500$. لاحظ كيف أن ١% تغيير فى معامل المتغير x_2 نتج عنه ١٠٠% تغيير فى الحل، وهذا بالطبع سيعتمد على مدى سوء تكيف التظام. لذلك فإنه فى الحالات التى نشك فيها فى سوء تكيف النظام فإننا نقوم بحساب محدد مصفوفة المعاملات ونقارنه بمقدار عناصر هذه المصفوفة لنقيم مدى سوء تكيف المصفوفة.

٥-٨ تمارين

١- استخدم طريقة جاوس العادية يدويا، ثم طريقة جاوس مع المحورة يدويا، ثم برنامج جاوس للحذف للتحقق من الحل لنظم المعادلات التالية واذكر أى فرق إن وجد:

(أ)
$$\begin{aligned} x_1 -5x_2 +x_3&=7\\ 10x_1 +20x_3&=6\\ 5x_1 -x_3&=4 \end{aligned}$$

(ب)
$$\begin{aligned} x_1 +x_2 -x_3&=1\\ x_1 +x_2 +4x_3&=2\\ 2x_1 -x_2 +2x_3&=3 \end{aligned}$$

(ت)
$$\begin{aligned} 2x_1 -3x_2 +2x_3&=5\\ -4x_1 +2x_2 -6x_3&=14\\ 2x_1 +2x_2 +4x_3&=8 \end{aligned}$$

(ث)
$$\begin{aligned} x_2 +x_3&=6\\ x_1 -2x_2 -x_3&=4\\ x_1 -x_2 +x_3&=5 \end{aligned}$$

٢- استخدم طريقة جاوس بدون ومع المحورة مع التقريب لأقرب ثلاث خانات عشرية لحل نظم المعادلات الخطية التالية يدويا وتحقق باستخدام البرنامج وقارن مع الحل الصحيح الموضح:

(أ)
$$\begin{aligned} 0.03x_1+58.9x_2&=59.2\\ 5.31x_1-6.10x_2&=47.0 \end{aligned}$$

الحل الصحيح هو $x_1=10$ و $x_2=1$.

(ب)
$$58.9x_1+0.03x_2=59.2$$
$$-6.10x_1+5.31x_2=47.0$$

الحل الصحيح هو $x_1=1$ و $x_2=10$.

(ت)
$$3.03x_1-12.1x_2+14x_3=-119$$
$$-3.03x_1+12.1x_2-7x_3=120$$
$$6.11x_1-14.2x_2+21x_3=-139$$

الحل الصحيح هو $x_1=0$ و $x_2=10$ و $x_3=1/7$.

٣- أعد التمرين السابق مستخدما التقريب لخمس خانات عشرية.

٤- عدل برنامج خواريزم جاوس الذى فى هذا الفصل بحيث يمكن استخدامه مع أى عدد من متجهات الثوابت b. أى يتم إدخال مصفوفة المعاملات، ثم متجهات الثوابت وعددها، ويعطى البرنامج مجموعة من متجهات الحلول مساوية لعدد متجهات الثوابت.

٥- صنف المصفوفات التالية كمصفوفات متفردة، أو سيئة التكيف، أو جيدة التكيف عن طريق حساب محدد كل منها:

(أ)
$$A=\begin{pmatrix}1&2&3\\2&3&4\\3&4&5\end{pmatrix}$$

(ب)
$$A=\begin{pmatrix}2.11&-0.80&1.72\\-1.84&3.03&1.29\\-1.57&5.25&4.30\end{pmatrix}$$

(ت)
$$A=\begin{pmatrix}2&-1&0\\-1&2&-1\\0&-1&2\end{pmatrix}$$

(ث)
$$A=\begin{pmatrix}4&3&-1\\7&-2&3\\5&-18&13\end{pmatrix}$$

٦- استخدم طريقة جاوس للحذف مع المحورة لحل النظام المتعدد التالى:

$$A=\begin{pmatrix}2&0&-1&0\\0&1&2&0\\-1&2&0&1\\0&0&1&-2\end{pmatrix}\qquad B=\begin{pmatrix}1&0\\0&0\\0&1\\0&0\end{pmatrix}$$

٧- حدد معاملات كثيرة الحدود التالية والتى تمر عبر النقاط (0, 10) و (1, 35) و (3, 31) و (4, 2):

$$y=a_0+a_1x+a_2x^2+a_3x^3$$

٨- حدد كثيرة الحدود من الدرجة الرابعة التى تمر خلال النقاط التالية: (0,-1) و (1, 1) و (3, 3) و (5, 2) و (6, -2).

الفصل ٦

الطرق التكرارية iterative لحل نظم المعادلات الخطية

الفصل ٦

الطرق التكرارية iterative لحل نظم المعادلات الخطية

طرق الحل لأى نظام من المعادلات الخطية التى رأيناها فى الفصل الخامس تعتبر طرقا مباشرة لإيجاد الحل، وتتميز هذه الطرق بأنها تصل إلى الحل النهائى بعدد محدد من الخطوات، وحيث أن الحاسب يعمل حسب دقة محددة نتيجة العدد المحدد لخانات تمثيل الأرقام فإن الحل الناتج لابد سيكون به خطأ معين سيتوقف على الطريقة المستخدمة، ولو فرض جدلا أن الحاسب يمكنه أن يعمل بدقة متناهية فى تمثيل الأرقام لتوصلت هذه الطرق إلى الحل الصحيح تماما.

على العكس من ذلك، فإن الطرق التكرارية التى نقدمها هنا، والتى تسمى بالطرق غير المباشرة، تبدأ بتخمين حل معين، وبطريقة تكرارية تحاول هذه الطرق التحسين من هذا الحل الذى تم تخمينه فى خطوات متتالية حتى يصبح التغير فى الحل مع التقدم فى هذه الخطوات صغيرا، حيث عندها نوقف الخطوات ويكون هذا الحل الأخير هو الحل المقدم لمجموعة المعادلات. هذه الخطوات التكرارية من الممكن أن يكون عددها كبيرا جدا، ولذلك فإن هذه الطرق تكون أبطأ كثيرا من الطرق المباشرة. الجدير بالذكر أن هذه الطرق لا تعتمد على قيمة الحل الابتدائى الذى نبدأ به محاولات الحل. كما أن هذه الطرق من الممكن ألا تصل إلى حل، بمعنى أن الحلول تتباعد مع تكرار المحاولات. على الرغم من ذلك فإن هذه الطرق يكون لها المميزات التالية التى تجعلها هى الأفضل عند حل مسائل معينة:

١- فى المصفوفات المتناثرة العناصر Sparse تكون معظم عناصر المصفوفة تساوى أصفارا، ولذلك فإن هذه الطرق تسمح بتخزين العناصر غير الصفرية فقط والتعامل معها حسابيا مما يوفر وقتا فى الحساب وفى مساحة التخزين وذلك على العكس من الطرق المباشرة التى لا تميز بين العناصر الصفرية وغير الصفرية فى الحساب. وهناك الكثير من التطبيقات التى تكون مصفوفاتها متناثرة.

٢- الميزة الثانية أن الطريقة التكرارية تكون ذاتية التصحيح، بمعنى أن كل محاولة للحل تكون أصح من المحاولة السابقة لها.

٣- الميزة الثالثة أن الطرق التكرارية لا تغير من قيم عناصر المصفوفة الأصلية، على العكس من الطرق المباشرة فإنها تغير من قيم عناصر المصفوفة الأصلية بل وفى ترتيب الصفوف كما رأينا فى الفصل السابق.

٤- فى الطرق المباشرة لابد من السير فيها حتى نهايتها وبعد ذلك نحسب الخطأ الناتج، أما فى الطرق التكرارية فإنه بعد كل محاولة يتم حساب الخطأ وعند وصول الخطأ إلى قيمة مناسبة يتوقف البرنامج، ولذلك من الممكن توفير الكثير من الوقت اللازم للحل.

من عيوب الطرق التكرارية أنها قد لا تصل إلى حل، بمعنى أن الحلول تتباعد مع تكرار الخطوات، والمثير أنه إذا تباعدت الحلول مع البداية بحل ابتدائى معين، فإنه مع حل ابتدائى آخر ستتقارب الحلول وتصل الطريقة إلى حل نهائى. هذه الطرق تكون مضمونة الوصول إلى حل إذا كانت المصفوفة مسيطرة قطريا، بمعنى أن معظم عناصرها متركزة حول القطر. ويمكن تعريف المصفوفة A بأنها مسيطرة قطريا إذا كان فى أى صف فيها يكون مقدار العنصر القطرى أكبر من أو يساوى مجموع عناصر هذا الصف الأخرى، ويمكن التعبير عن ذلك بالمعادلة التالية:

$$|a_{ii}| = \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \qquad \text{(٦-١)}$$

وذلك لكل قيم i

٦-١ طريقة جاكوبى Jacobi التكرارية

لتوضيح هذه الطريقة، والطرق التكرارية على وجه العموم، سنفترض حل المعادلة البسيطة 3x+1=0 والتى حلها هو ببساطة x=(-1/3)، ولكننا سنطبق عليه الطريقة التكرارية لنرى كيفية تطبيق هذه الطرق.

المعادلة السابقة يمكن وضعها على الصورة التالية:

$$2x=-x-1$$

ومنها يمكن كتابة x كما يلى:

$$x=-0.5x-0.5$$

أو بالصورة التكرارية التالية:

$$x_{k+1}=-0.5x_k-0.5$$

ما حدث هنا هو أننا بصورة أو أخرى نحول المعادلة إلى x فقط فى الطرف الأيسر وباقى المعادلة فى الطرف الأيمن. الصورة التكرارية للمعادلة السابقة تعنى أن x عند أى مرحلة k+1 تساوى x عند المرحلة السابقة k مضروبة فى 0.5- ومجموعا عليها 0.5-. ويمكن تكرار ذلك فى خطوات متتالية كما يلى:

$$x_1=-0.5-0.5x_0$$

$$x_2=-0.5-0.5x_1=-0.5+(0.5)^2+(0.5)^2x_0$$

$$x_3=-0.5-0.5x_2=-0.5+(0.5)^2-(0.5)^{-3}-(0.5)^3x_0$$

وهكذا ....

كما نلاحظ فإنه مهما كانت قيمة x_0، فإن هذه العملية ستؤول إلى متوالية هندسية أساسها هو 0.5- وبالتالى ستصبح x_k كالتالى:

$$x_k = \frac{-1/2}{1-(-\frac{1}{2})} = -\frac{1}{3}$$

وهو الحل الصحيح للمعادلة. ولكن الأمر هنا ليس بهذه السهولة ولا يمكن تعميمه. أنظر مثلا لهذه المعادلة:

$$x=-2x-1$$

ومنها:
$$x_{k+1}=-2x_k-1$$

والتى يمكن كتابتها كما يلى:
$$x_1=-1-2x_0$$
$$x_2=-1-2x_1=-1-2(-1-2x_0)=-1+2+2^2x_0$$
$$x_3=-1-2x_2=-1+2-2^2-2^3x_0$$

وهكذا ......

أى أن المجموع يتباعد بصرف النظر عن قيمة x_0، وبالتالى فهذه الطريقة لن تؤدى إلى نتيجة.

دعنا الآن نفترض الحالة الأكثر عمومية وهى نظام من المعادلات الخطية كالتالى:

$$a_{11}x_1+a_{12}x_2+\ \dots\ +a_{1n}x_n=b_1$$
$$a_{21}x_1+a_{22}x_2+\ \dots\ +a_{2n}x_n=b_2$$
$$\dots\dots\dots\dots\dots\dots\dots\dots$$
$$a_{n1}x_1+a_{n2}x_2+\ \dots\ +a_{nn}x_n=b_n$$

ويجب أن نتذكر هنا أن مصفوفة المعاملات A تكون مصفوفة غير متفردة (محددها لا يساوى الصفر).

تبدأ طريقة جاكوبى بجعل الطرف الأيسر فى كل المعادلات السابقة يحتوى x فقط كما يلى:

$$x_1=(b_1-a_{12}x_2-a_{13}x_3-\ \dots\ -a_{1n}x_n)/a_{11}$$
$$x_2=(b_2-a_{21}x_1-a_{23}x_3-\ \dots\ -a_{2n}x_n)/a_{22}$$
$$\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$
$$x_n=(b_n-a_{n1}x_1-a_{n3}x_3-\ \dots\ -a_{n,n-1}x_{n-1})/a_{nn}$$

من ذلك نرى أنه حيث تتم القسمة دائما على العناصر القطرية a_{11} حتى a_{nn} فإنه يجب ألا يكون أى واحد من هذه العناصر يساوى صفر، وإذا حدث وكان أى من هذه العناصر يساوى صفرا فلابد من تبديل المعادلات حتى تكون كل العناصر القطرية لا تساوى الصفر، وإذا تعذر ذلك فإن ذلك يعنى أنه لن يوجد حل لهذه المعادلات لأن المصفوفة A متفردة.

يمكن كتابة الصورة العامة للمتجه x من المعادلات السابقة كما يلى:

$$x_i=\left(b_i-\sum_{\substack{j=1\\ j\neq i}}^{n}a_{ij}x_j\right)/a_{ii} \qquad (٦-٢)$$

حيث i تتغير من 1 إلى n و a_{ii} لا تساوى الصفر لكل قيم i.

ويمكن كتابة المعادلة (٦- ٢) فى صورة تكرارية كما يلى:

$$x_i^{(k+1)} = \left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j^k \right) / a_{ii} \quad \text{حيث} \quad k=0,1,2,\ldots \qquad (٦-٥)$$

حيث k تمثل مرات التكرار. هذه الطريقة ستتقارب إلى حل أكيد إذا كانت المصفوفة A مسيطرة قطريا، أى أن عناصرها تتركز حول القطر، أما إذا كانت غير ذلك فإن الوصول للحل لا يكون أكيدا، ولكنه ممكن أحيانا.

مثال ٦-١

إفترض نظام المعادلات التالى والمطلوب إيجاد حل له باستخدام طريقة جاكوبى:

$$\begin{pmatrix} 10x_1 - x_2 + 2x_3 = 6 \\ -x_1 + 11x_2 - x_3 + 3x_4 = 25 \\ 2x_1 - x_2 + 10x_3 - x_4 = -11 \\ 3x_2 - x_3 + 8x_4 = 15 \end{pmatrix} \qquad (٦-٦)$$

هذا النظام له الحل $x=(1, 2, -1, 1)^t$ حيث t تعنى دوران المتجه transpose ليصبح متجه عمود بدلا من متجه صف. الآن بتنفيذ الخطوة الأولى فى طريقة جاكوبى بجعل x فقط فى الطرف الأيسر نحصل على المعادلة التالية:

$$\begin{aligned} x_1 &= \frac{3}{5} + \frac{1}{10}x_2 - \frac{1}{5}x_3 \\ x_2 &= \frac{25}{11} + \frac{1}{11}x_1 + \frac{1}{11}x_3 - \frac{3}{11}x_4 \\ x_3 &= -\frac{11}{10} - \frac{1}{5}x_1 + \frac{1}{10}x_2 + \frac{1}{10}x_4 \\ x_4 &= \frac{15}{8} - \frac{3}{8}x_2 + \frac{1}{8}x_3 \end{aligned} \qquad (٦-٧)$$

بوضع الحل المبدئى $x^{(0)} = (0.0.0.0)^t$ والتعويض فى المعادلة (٦- ٧) نحصل على $x^{(1)}$ كما يلى:

$$\begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \\ x_4^{(1)} \end{pmatrix} = \begin{pmatrix} 0.6000 \\ 2.2727 \\ -1.1000 \\ 1.8750 \end{pmatrix} \qquad (٦-٨)$$

بالاستمرار فى حساب $x^{(2)}$ و $x^{(3)}$ و ....$x^{(10)}$ وهكذا نحصل على الجدول التالى بعد عشرة محاولات:

	x_1	x_2	x_3	x_4
k=1	0.6000	2.2727	-1.1000	1.8750
k=2	1.0473	1.7159	-0.8052	0.8852
k=3	0.9326	2.0533	-1.0493	1.1309
k=4	1.0152	1.9537	-0.9681	0.9738
k=5	0.9890	2.0114	-1.0103	1.0214
k=6	1.0032	1.9922	-0.9945	0.9944

k=7	0.9981	2.0023	-1.0020	1.0036
k=8	1.0006	1.9987	-0.9990	0.9989
k=9	0.9997	2.0004	-1.0004	1.0006
k=10	1.0001	1.9998	-0.9998	0.9998

البرنامج المقترح لطريقة جاكوبى هو كما يلى:

```
1- %program to solve a system of linear equations using Jacobi method
2- clear ;
3- format short
4- a = input([' Enter The Array of Coefficients a= ']); %e.g [1 2;3 4]
5- b = input([' Enter The Arrays of Constants b = ' ]) ;%e.g [1;2;]
6- n = size(a,1) ;% where a is nxn and we want n only
7- x0 = input([' Enter The Array of first trial x0 = ' ]) ;%e.g [1;2;]
8- kmax = input([' Enter The Max number of iterations = ' ]) ;%e.g [10]
9- X = x0; At = zeros(n,n);
10-      for i = 1:n
11-         for j = 1:n
12-            if j ~= i
13-               At(i,j) = -a(i,j)/a(i,i);
14-            end
15-          end
16-         Bt(i,:) = b(i,:)/a(i,i);
17-      end
18-      for k = 1: kmax
19-          X = At*x0 + Bt;
20-           x0 = X;
21-      end
22-      disp(X);
```

وكانت نتيجة تنفيذ هذا البرنامج لطبع آخر متجه فقط كالتالى:

```
>> Jacobi
Enter The Array of Coefficients a= [10 -1 2 0;-1 11 -1 3;2 -1 10 -1;0 3 -1 8]
Enter The Arrays of Constants b = [6;25;-11;15]
Enter The Array of first trial x0 = [0;0;0;0]
Enter The Max number of iterations = 10
  1.0001
  1.9998
 -0.9998
  0.9998
```

فى البرنامج السابق تستخدم الأوامر من ١ إلى ٨ لإدخال الثوابت التى منها مصفوفة المعاملات A ومتجه الثوابت b ومتجه الحلول الابتدائى x0 الذى سنبدأ به عملية التكرار، وكذلك أكبر عدد للخطوات التكرارية التى سيسمح بها،

والخطأ أو السماح بين محاولتين تاليتين بحيث إذا كان هذا الخطأ أقل من كمية معينة تتوقف عملية التكرار. لاحظ أنه لم يتم إدراج هذا الخطأ وتم تركه كتمرين يقوم به القارىء فى نهاية الفصل.

الأمر رقم ٩ قام بتصفير مصفوفة At ليتم وضع نتائج المعالجة فيها وكذلك تم وضع X=x0 بحيث أن X سيكون هو متجه الحل الذى سيستخدم داخل الحلقة. بعد ذلك تم الدخول فى حلقة i على صفوف المصفوفة، وفى داخل هذه الحلقة توجد حلقة أخرى j على الأعمدة فى كل صف ما عدا العمود الذى يحتوى العنصر القطرى، وذلك لتنفيذ الخطوة الأولى من خواريزم جاكوبى بجعل x فقط فى الطرف الأيسر مع قسمة باقى المعادلات على العناصر القطرية. هاتين الحلقتين أخذتا الأوامر من ١٠ إلى ١٧. الأوامر من ١٨ حتى ٢١ دخلت فى حلقة حساب متجه الحلول الجديد X بمعرفة المتجه x_0 والتعويض فى معادلة النظام $X=At*x_0+Bt$ فى الأمر ١٩. بعد حساب المتجه X نضعه فى x_0 كما فى الأمر ٢٠ لنبدأ حلقة جديدة لحساب قيمة جديدة للمتجه X. ثم تنتهى الحلقة وبالتالى البرنامج ويتم عرض النتائج بالطريقة التى تناسبك.

٦-٢ طريقة جاوس سايدل Gauss Seidel التكرارية

هذه الطريقة تعتبر تعديل لطريقة جاكوبى ولكنها وإن كانت فكرتها بسيطة إلا أنها تؤدى فى بعض الأحيان إلى تسريع كبير جدا فى تنفيذ البرنامج. لكى نفهم هذه الطريقة سنكتب معادلة متجه المجاهيل فى طريقة جاكوبى بعد نقل المجاهيل للطرف الأيسر هنا مرة ثانية للتذكرة:

$$x_i^{(k+1)} = \left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j^k\right)/a_{ii} \quad \text{حيث } k=0,1,2,\dots \qquad (٦-٩)$$

هذه المعادلة تحسب قيمة العنصر x_i من عناصر مصفوفة المجاهيل x عند المحاولة k+1 بدلالة عنصر متجه الثوابت b_i وعناصر باقى المجاهيل x_j حيث j هى عناصر الأعمدة التى فى نفس الصف i ما عدا العنصر القطرى a_{ii}. هذه المعادلة السابقة يمكن إعادة كتابتها على الصورة التالية:

$$x_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^k - \sum_{j=i+1}^{n} a_{ij}x_j^k\right)/a_{ii} \qquad (٦-١٠)$$

المعادلة (٦- ١٠) هى نفسها المعادلة (٦- ٩) سوى أننا قسمنا المقدار الذى تحت علامة المجموع إلى مقدارين، المقدار الأول هو مجموع عناصر الصف i التى قبل العنصر القطرى a_{ii} (j تتغير من 1 حتى i-1)، والمقدار الثانى هو مجموع عناصر الصف i أيضا التى بعد العنصر القطرى a_{ii} (j تتغير من i+1 حتى n). السؤال الذى يتبادر إلى الذهن فجأة هو ما هى الفائدة من وراء هذه القسمة؟ الفائدة عظيمة كما سنرى.

فى الحقيقة أن ما يحدث هو أن حساب عناصر مصفوفة المجاهيل x يتم تتابعيا، بمعنى أننا عند المحاولة k+1 نقوم بحساب $x_1^{(k+1)}$ ثم بعدها نحسب $x_2^{(k+1)}$ ثم $x_3^{(k+1)}$ وهكذا. إذن معنى ذلك أننا عندما نكون وصلنا إلى حساب العنصر $x_i^{(k+1)}$ فإننا نكون قد حسبنا كل عناصر المجاهيل من $x_1^{(k+1)}$ حتى $x_{i-1}^{(k+1)}$ وهى المجموع الأول فى المعادلة (٦-١٠). إذن لماذا لا نستخدم هذه القيم الجديدة (قيم المحاولة k+1) فى المعادلة (٦- ١٠) بدلا من الاستمرار بالتعويض بقيم المحاولة k. بالتأكيد فإن ذلك سيعمل على تسريع خوارزيم جاكوبى كما سنرى. وعلى ذلك فالمعادلة (٦- ١٠) ستصبح كالتالى:

$$x_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij} x_j^{k}\right) / a_{ii} \qquad (٦-١١)$$

البرنامج التالى يمثل تعديل لبرنامج جاكوبى السابق ليعطى الحل بطريقة جاوس سايدل:

```
% solving a system of linear equations using Gauss-Siedel method
clear ;
format short
a = input([' Enter The Array of Coefficients a= ']); %e.g [1 2;3 4]
b = input([' Enter The Arrays of Constants b = ' ]) ;%e.g [1;2;]
n = size(a,1) ;% where a is nxn and we want n only
x0 = input([' Enter The Array of first trial x0 = ' ]) ;%e.g [1;2;]
kmax = input([' Enter The Max number of iterations = ' ]) ;%e.g [10]
X = x0; At = zeros(n,n);
for k = 1: kmax
    X(1,:) = (b(1,:)-a(1,2:n)*x0(2:n,:))/a(1,1);
    for i = 2:n-1
       tmp = b(i,:)-a(i,1:i-1)*X(1:i-1,:)-a(i,i+1:n)*x0(i+1:n,:);
       X(i,:) = tmp/a(i,i);
    end
    X(n,:) = (b(n,:)-a(n,1:n - 1)*X(1:n - 1,:))/a(n,n);
    x0 = X;
    disp(X');
end
%disp(X);
```

مثال ٦-٢

بتنفيذ هذا البرنامج على نفس مجموعة المعادلات السابقة التى استخدمناها مع برنامج جاكوبى وبنفس عدد المحاولات لنقارن النتائج نحصل على ما يلى:

Enter The Array of Coefficients a= [10 -1 2 0;-1 11 -1 3;2 -1 10 -1;0 3 -1 8]

Enter The Arrays of Constants b = [6;25;-11;15]

Enter The Array of first trial x0 = [0;0;0;0]

Enter The Max number of iterations = 10

	x_1	x_2	x_3	x_4
1-	0.6000	2.3273	-0.9873	0.8789
2-	1.0302	2.0369	-1.0145	0.9843
3-	1.0066	2.0036	-1.0025	0.9984
4-	1.0009	2.0003	-1.0003	0.9998
5-	1.0001	2.0000	-1.0000	1.0000
6-	1.0000	2.0000	-1.0000	1.0000
7-	1.0000	2.0000	-1.0000	1.0000
8-	1.0000	2.0000	-1.0000	1.0000
9-	1.0000	2.0000	-1.0000	1.0000
10-	1.0000	2.0000	-1.0000	1.0000

لقد نفذنا البرنامج هنا على ١٠ محاولات مثل طريقة جاكوبى تماما والجدير بالملاحظة أننا حصلنا على الإجابة الصحيحة تماما لمتجه المجاهيل x بعد ٥ محاولات فقط كما هو موضح فى النتائج السابقة التى حصلنا عليها، وهذا يعكس الفرق فى السرعة بين هذه الطريقة وطريقة جاكوبى.

مثال ٦-٣

استخدم طريقة جاوس سايدل لحل مجموعة المعادلات التالية مع العلم أن الحل الصحيح هو $x^T=[1\ 1\ 1\ 1]$:

$$\begin{pmatrix} 5 & 7 & 6 & 5 \\ 7 & 10 & 8 & 7 \\ 6 & 8 & 10 & 9 \\ 5 & 7 & 9 & 10 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}=\begin{pmatrix} 23 \\ 32 \\ 33 \\ 31 \end{pmatrix}$$

بالتعويض فى برنامج جاوس سايدل السابق عند أعداد محاولات k مختلفة حصلنا على مايلى:

	x_1	x_2	x_3	x_4
k=50	2.2033	0.2767	0.6947	1.1795
k=100	2.0302	0.3808	0.7384	1.1537
k=500	1.2973	0.8213	0.9245	1.0444
k=1000	1.0629	0.9622	0.9840	1.0094
k=10000	1.0000	1.0000	1.0000	1.0000

نلاحظ أننا وصلنا للحل الصحيح ولكن بعد عشرة آلاف محاولة تقريبا، أو التقارب للحل الصحيح كان بطيئا جدا.
السؤال الآن ما هو موقف هذه المسألة عند حلها بطريقة جاكوبى، وهذا ما سنراه فى المثال التالى:

مثال ٦–٤

طبق طريقة جاكوبى على نفس المثال السابق وعند نفس عدد المحاولات k.

		x_1	x_2	x_3	x_4
k=50	1.0e+19 *	[-6.4874	-4.5145	-4.5759	-4.2500]
k=100	1.0e+39 *	[-3.1461	-2.1893	-2.2191	-2.0611]
k=500	1.0e+196 *	[-9.6249	-6.6979	-6.7890	-6.3055]
k=1000		NaN	NaN	NaN	NaN

نلاحظ أننا لم نصل هنا إلى حل حيث رأينا أن متجه المجاهيل يتباعد باستمرار (تزداد قيمته) إلى أن وصلنا إلى القيمة NAN والتى تعنى رقم لا نهائى.

أى أن هذا النظام من المعادلات لم نصل إلى حل له باستخدام طريقة جاكوبى، وتم الوصول إلى حل بطىء جدا باستخدام طريقة جاوس سايدل، وكان من الممكن أن لا نصل إلى حل أيضا مع طريقة جاوس سايدل. نلاحظ فى هذا النظام للمعادلات أنه ليس مسيطر قطريا diagonally dominant، أى أن القيمة المطلقة لأى عنصر قطرى فى أى صف أقل من مجموع باقى العناصر فى باقى الصفوف، لذلك فإن حل هذا النظام بأى واحدة من الطريقتين السابقتين لا يكون مضمونا.

مثال ٦–٥

المثال التالى يوضح نظام مسيطر قطريا وسنرى كيف ستتعامل معه كل من طريقتى جاكوبى وطريقة جاوس سايدل مع ملاحظة أن الحل الصحيح لهذا النظام هو $x^T=[1\ 1\ 1]$.

$$\begin{pmatrix} 3 & 1 & -1 \\ 2 & 6 & 2 \\ 1 & 2 & 4 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 10 \\ 7 \end{pmatrix}$$

أولا باستخدام طريقة جاكوبى باستخدام عدد من المحاولات k يساوى ١٢ حصلنا على مايلى:

	x_1	x_2	x_3
k=1	1.0000	1.6667	1.7500
k=2	1.0278	0.7500	0.6667
k=3	0.9722	1.1019	1.1181
k=4	1.0054	0.9699	0.9560
k=5	0.9954	1.0129	1.0137
k=6	1.0003	0.9970	0.9947
k=7	0.9992	1.0017	1.0014
k=8	0.9999	0.9998	0.9994

k=9	0.9999	1.0002	1.0001
k=10	1.0000	1.0000	0.9999
k=11	1.0000	1.0000	1.0000
k=12	1.0000	1.0000	1.0000

حيث نلاخظ أن متجه المجاهيل استقر على الحل الصحيح بعد ١١ محاولة.

ثانيا باستخدام طريقة جاوس سايدل:

	x_1	x_2	x_3
k=1	1.0000	1.3333	0.8333
k=2	0.8333	1.1111	0.9861
k=3	0.9583	1.0185	1.0012
k=4	0.9942	1.0015	1.0007
k=5	0.9997	0.9999	1.0001
k=6	1.0001	0.9999	1.0000
k=7	1.0000	1.0000	1.0000
k=8	1.0000	1.0000	1.0000

حيث نلاحظ أن النظام قد وصل إلى الحل الصحيح بعد ٧ محاولات.

نخلص من ذلك أنه عندما يكون نظام المعادلات مسيطر قطريا diagonally dominant فإنه بالتأكيد سيصل إلى حل باستخدام أى واحدة من الطريقتين ولكن طريقة جاوس سايدل ستكون أسرع من طريقة جاكوبى. ولكن إذا كان النظام غير مسيطر قطريا فإن الوصول إلى حل بأى من الطريقتين يكون غير مؤكد، ولكن فرصة الوصول لحل ربما تكون أفضل مع جاوس سايدل. تذكر أيضا أن الحل الأولى الذى نبدأ به عملية التكرار يؤثر فقط فى سرعة الوصول إلى الحل الصحيح، ولكنه لا يكون له تأثير على التقارب للحل أو عدم التقارب.

كلمة أخيرة نذكرها هنا عن الفرق بين طريقة جاكوبى وطريقة جاوس سايدل هى أن التعامل المتتالى لطريقة جاكوبى مع مصفوفة المعاملات يجعلها أسهل فى المواءمة للحساب المتوازى على الحاسبات متعددة النويات multiprocessors بينما طريقة جاوس سايدل تكون أصعب فى ذلك.

٦–٣ طريقة الاسترخاء المفرط المتوالى SOR التكرارية

هذه الطريقة (طريقة الاسترخاء المفرط successive over relaxation, SOR) عبارة عن تطوير لطريقة جاوس سايدل التكرارية. فى هذه الطريقة يتم إعادة ترتيب المعادلة (٦– ١١) الخاصة بطريقة جاوس سايدل بحيث تكون المحاولة

$x_i^{(k+1)}$ لحساب عناصر متجه المجاهيل تساوى المحاولة $x_i^{(k)}$ زائد كمية تصحيحية يمكن التحكم فيها عن طريق معامل استرخائى ω يمكن التحكم فى قيمته كما يلى:

$$x_i^{(k+1)} = x_i^{(k)} + \omega r_i^{(k)} \qquad (١٢-٦)$$

لكى نصل إلى هذه الصورة ونوجد $r_i^{(k)}$ فإننا سنجمع ونطرح نفس الكمية $x_i^{(k)}$ على معادلة طريقة جاوس سايدل، المعادلة (٦– ١١) ، وبذلك يمكن إعادة كتابتها كما يلى:

$$x_i^{(k+1)} = x_i^k - x_i^k + \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}x_j^k\right)/a_{ii} \qquad (١٢-٦)$$

الكمية $-x_i^k$ يمكن إدخالها القوس وفى داخل علامة المجموع اليمنى لتصبح المعادلة كما يلى:

$$x_i^{(k+1)} = x_i^k + \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i}^{n} a_{ij}x_j^k\right)/a_{ii} \qquad (١٣-٦)$$

لاحظ أن حدود علامة الجمع الثانية أصبحت حدودها بحيث أن j تتغير من j=i بدلا من j=i+1 لكى تستوعب الكمية x_i^k المضافة بداخلها. بذلك فإن r_i^k يمكن كتابتها كما يلى:

$$r_i^{(k)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i}^{n} a_{ij}x_j^k\right)/a_{ii} \qquad (١٤-٦)$$

بذلك يمكن كتابة الصورة النهائية لطريقة الاسترخاء المتتالى SOR كما يلى:

$$x_i^{(k+1)} = x_i^k + \omega\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i}^{n} a_{ij}x_j^k\right)/a_{ii} \qquad (١٥-٦)$$

لاحظ أنه بوضع معامل الاسترخاء ω بواحد فإن المعادلة (٦– ١٥) تؤول إلى معادلة جاوس سايدل الأصلية.

بوضع المعامل ω خارج المدى 0<ω<2 فإن طريقة الاسترخاء المتتالى لن تتقارب إلى حل نهائى. بوضع ω فى المدى 1<ω<2 فإن التقارب يكون أسرع من جاوس سايدل فى المسائل التى تتقارب أصلا مع طريقة جاوس سايدل، وبوضع ω فى المدى 0<ω<1 فإنه قد يمكن التقارب إلى حل فى المسائل التى لا تتقارب مع طريقة جاوس.

البرنامج التالى يبين التعديلات على برنامج طريقة جاوس سايدل ليتلائم مع طريقة الاسترخاء المتتالى SOR:

```
%program to solve a system of linear equations by the SOR method
clear ;
format short
a = input([' Enter The Array of Coefficients a= ']); %e.g [1 2;3 4]
b = input([' Enter The Arrays of Constants b = ' ]) ;%e.g [1;2;]
n = size(a,1) ;% where a is nxn and we want n only
x0 = input([' Enter The Array of first trial x0 = ' ]) ;%e.g [1;2;]
kmax = input([' Enter The Max number of iterations = ' ]) ;%e.g [10]
w = input([' Enter The Relaxation Value = ' ]) ;%e.g [10]
X = x0; At = zeros(n,n);
for k = 1: kmax
    X(1,:) = x0(1,:)+w*((b(1,:)-a(1,1:n)*x0(1:n,:)))/a(1,1);
    for i = 2:n-1
        tmp = w*(b(i,:)-a(i,1:i-1)*X(1:i - 1,:)-a(i,i:n)*x0(i:n,:));
        X(i,:) = x0(i,:)+tmp/a(i,i);
```

```
    end
    X(n,:) = x0(n,:)+w*((b(n,:)-a(n,1:n)*X(1:n,:)))/a(n,n);
    x0 = X;
    disp(X');
end
%disp(X');
```

سنختبر هذا البرنامج على نظام المعادلات الموضح فى مثال ٦-٦ التالى:

مثال ٦-٦

أوجد حل نظام المعادلات التالى باستخدام طريقة جاوس سايدل وطريقة الاسترخاء المتتالى SOR مع العلم بأن الحل الصحيح لهذا النظام هو x=[3;4;-5]:

$$\begin{pmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 24 \\ 30 \\ -24 \end{pmatrix}$$

أولا باستخدام طريق جاوس سايدل ومتجه مجاهيل ابتدائى x=[1;1;1] نحصل على النتائج التالية بعد عدد محاولات kmax=7:

	x_1	x_2	x_3
k=1	5.5200	3.8125	-5.0469
k=2	3.1406	3.8828	-5.0293
k=3	3.0879	3.9268	-5.0183
k=4	3.0549	3.9542	-5.0114
k=5	3.0343	3.9714	-5.0072
k=6	3.0215	3.9821	-5.0045
k=7	3.0134	3.9888	-5.0028

ثانيا باستخدام طريقة الاسترخاء المتتالى SOR مع ω=1.25 ونفس متجه البداية للمجاهيل نحصل على ما يلى:

	x_1	x_2	x_3
k=1	6.3125	3.5195	-6.6501
k=2	2.6223	3.9585	-4.6004
k=3	3.1333	4.0103	-5.0967
k=4	2.9571	4.0075	-4.9735
k=5	3.0037	4.0029	-5.0057
k=6	2.9963	4.0009	-4.9983
k=7	3.0000	4.0003	-5.0003

نلاحظ أنه عند المحاولة السابعة كان الخطأ مع طريقة الاسترخاء أقل بكثير عن نظيره مع طريقة جاوس سايدل. لاحظ أيضا بأن هذا التحسين سيتوقف على مدى السيطرة القطرية لمصفوفة المعاملات، وعلى قيمة معامل الاسترخاء ω. ولذلك لو أعدنا نفس المثال السابق باستخدام ω=1.7 نحصل على ما يلى:

	x_1	x_2	x_3
k=1	8.2250	1.9881	-10.0550
k=2	1.9076	4.6527	-1.1841
k=3	2.9325	5.2510	-7.1395
k=4	1.4523	4.1884	-3.4223
k=5	3.8432	3.4636	-6.3324
k=6	3.0937	3.6898	-4.1992
k=7	3.3300	4.1368	-5.5024

لاحظ مدى الفرق فى النتائج لمجرد تكبير قيمة معامل الاسترخاء. على الفور من الممكن أن يظهر هنا سؤال وهو كيفية اختيار القيمة المثلى لهذا المعامل. فى الحقيقة لا توجد إجابة قاطعة لهذا السؤال سوى التجربة والخطأ وإن وجدت بعض المحاولات لوضع معادلة لهذا المعامل ولكنها كلها ليست قاطعة.

فى كل الأمثلة السابقة كنا نستخدم عدد معين من المحاولات ونرى هل وصل البرنامج إلى الحل الصحيح أم لا بناء على أننا كنا نعرف الحل الصحيح فى كل حالة. فى المواقف التطبيقية لا يكون لدينا معرفة بالحل الصحيح لذلك فإننا ننفذ البرنامج ونحسب الخطأ بين كل محاولة والسابقة لها ونتوقف عند وصول هذا الخطأ إلى قيمة مناسبة. السؤال هنا هو كيف يمكن حساب الخطأ بين أى متجهين؟ فى هذه الحالة يتم استخدام معيار المتجه، vector norm وهو يمثل القيمة المطلقة للمتجه التى يمكن كتابتها كما يلى:

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \cdots . + x_n^2} \qquad (٦-١٦)$$

هذا المعيار يمثل المسافة الإكلودية بين نقطة الأصل والنقطة التى يمثلها هذا المتجه فى الفراغ. يمكن استخدام هذا التعبير عن الخطأ فى كل البرامج السابقة.

٦-٤ تمارين

١- نفذ أول ثلاث خطوات من طريقة جاكوبى التكرارية لحل النظم الخطية التالية مستخدما متجه المجاهيل الابتدائى يساوى صفر:

(أ)

$$3x_1 - x_2 + x_3 = 1$$
$$3x_1 + 6x_2 + 2x_3 = 0$$
$$3x_1 + 3x_2 + 7x_3 = 4$$

(ب)
$$\begin{aligned} 10x_1 - x_2 \quad &= 9 \\ -x_1 + 10x_2 - 2x_3 &= 7 \\ -2x_2 + 10x_3 &= 6 \end{aligned}$$

(ت)
$$\begin{aligned} 4x_1 + x_2 + x_3 \quad + x_5 &= 6 \\ -x_1 - 3x_2 + x_3 + x_4 \quad &= 6 \\ 2x_1 + x_2 + 5x_3 - x_4 - x_5 &= 6 \\ -x_1 - x_2 - x_3 + 4x_4 \quad &= 6 \\ 2x_2 - x_3 + x_4 + 4x_5 &= 6 \end{aligned}$$

(ث)
$$\begin{aligned} 4x_1 - x_2 + \quad x_4 \quad &= 0 \\ -x_1 + 4x_2 - x_3 \quad - x_5 &= 5 \\ - x_2 + 4x_3 \quad - x_6 &= 0 \\ -x_1 \quad + 4x_4 - x_5 &= 6 \\ -x_2 \quad - x_4 + 4x_5 - x_6 &= -2 \\ -x_3 \quad - x_5 + 4x_6 &= 6 \end{aligned}$$

٢- أعد تمرين ١ مستخدما طريقة جاوس سايدل

٣- أعد تمرين ١ مستخدما طريقة الاسترخاء SOR بقيم مختلفة لمعامل الاسترخاء بدءا من $\omega=1.1$.

٤- برنامج طريقة جاكوبى مصمم لينفذ عدد معين من المحاولات ثم يطبع آخر نتيجة. عدل هذا البرنامج بحيث يطلب منك إدخال الخطأ أو السماحية بين محاولتين متتاليتين ويظل يدور فى المحاولات إلى أن يصل إلى هذه الدقة. بالطبع يتم إعطاؤه أيضا أكبر عدد من المحاولات kmax، بحيث إذا وصل إلى الدقة المطلوبة ولم يصل إلى kmax يخرج ويعطى رسالة توضح ذلك، وإذا خرج نتيجة وصوله إلى kmax دون الوصول إلى الدقة المطلوبة يعطى رسالة أخرى تدل على ذلك.

٥- أعد تمرين ٤ على طريقة جاوس سايدل.

٦- أعد تمرين ٤ على طريقة الاسترخاء SOR .

٧- نفذ تمرين ٤ و ٥ و ٦ على أحد المسائل وارسم تغير الخطأ مع عدد الدورات أو المحاولات فى نفس المخطط لتوضح الفرق فى تغير الخطأ مع عدد المحاولات للطرق الثلاث.

٨-
$$\begin{aligned} x_1 \quad - x_3 &= 0.2 \\ -0.5x_1 + x_2 - 0.25x_3 &= -1.425 \\ x_1 - 0.5x_2 + x_3 &= 2 \end{aligned}$$

هذا النظام للمعادلات الخطية له الحل $(0.9, -0.8, 0.7)^t$.

(أ) هل مصفوفة المعاملات لهذا النظام مسيطرة قطريا.

(ب) استخدم طريقة جاكوبى التكرارية لإيجاد حل تقريبى لهذا النظام بدقة 0.01 وعدد أكبر من المحاولات يساوى ٣٠٠.

(ت) ماذا يحدث للحل فى الجزء (ب) إذا تغير النظام إلى:

$$
\begin{aligned}
x_1 \qquad\quad -2x_3 &= 0.2 \\
-0.5x_1 + x_2 - 0.25x_3 &= -1.425 \\
x_1 - 0.5x_2 + x_3 &= 2
\end{aligned}
$$

٩- أعد تمرين ٨ مستخدما طريقة جاوس سايدل.

١٠- أعد ٨ مستخدما طريقة الاسترخاء SOR.

١١- استخدم الثلاث طرق جاكوبى وجاوس سايدل والاسترخاء لحل نظام المعادلات التالى:

$$
\begin{pmatrix} -2 & 5 & 9 \\ 7 & 1 & 1 \\ -3 & 7 & -1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 6 \\ -26 \end{pmatrix}
$$

١٢-
$$
\begin{pmatrix} 4 & -1 & 0 & 0 \\ -1 & 4 & -1 & 0 \\ 0 & -1 & 4 & -1 \\ 0 & 0 & -1 & 3 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 15 \\ 10 \\ 10 \\ 10 \end{pmatrix}
$$

حل هذا النظام مستخدما الطرق الثلاثة مع $x_i=b_i/a_{ii}$ كمتجه مجاهيل البتدائى و $\omega=1.1$ كمعامل استرخاء مع طريقة الاسترخاء.

١٣- نظام المعادلات التالى غير مسيطر قطريا حاول أن تحله باستخدام طريقة جاكوبى وجاوس سايدل مستخدما الحل الابتدائى $x=(0, 0, 0)$.

$$
\begin{aligned}
x_1 + 3x_2 - x_3 &= 5 \\
3x_1 - x_2 \qquad &= 5 \\
x_2 - 2x_3 &= 1
\end{aligned}
$$

١٤- فى تمرين ١٣ حاول إجراء بعض التبديلات على صفوف نظام المعادلات لتجعله مسيطر قطريا ثم حاول الحل باستخدام طريقة جاكوبى وجاوس سايدل.

١٥- أعد تمرين ١٣ وتمرين ١٤ لنظام المعادلات التالى:

$$
\begin{aligned}
2x_1 - 3x_2 \qquad &= -7 \\
x_1 + 3x_2 - 10x_3 &= 9 \\
3x_1 \qquad + 4x_3 &= 13
\end{aligned}
$$

١٦- حل نظام المعادلات الخطى التالى باستخدام طريقة جاكوبى وجاوس سايدل:

$$
\begin{aligned}
4x_1 - x_2 - x_3 \qquad &= 18 \\
-x_1 + 4x_2 - x_3 - x_4 &= 18
\end{aligned}
$$

$$-x_2+4x_3-x_4-x_5=4$$
$$-x_3+4x_4-x_5-x_6=4$$
$$-x_4+4x_5-x_6-x_7=26$$
$$-x_5+4x_6-x_7-x_8=16$$
$$-x_6+4x_7-x_8=10$$
$$-x_7+4x_8=32$$

الفصل ٧

الاستيفاء وتقريب المنحنيات

Interpolation and curve fitting

الفصل ٧

الاستيفاء وتقريب المنحنيات

Interpolation and curve fitting

يحتاج العلماء والمهندسون ومن يتعاملون مع البيانات أن يمثلوا العلاقة بين متغيرين أو أكثر تكون لديهم قراءات لهما. تخيل مثلا فى أحد العمليات الصناعية يتم تسجيل درجة الحرارة على مدار الساعة كل نصف ساعة فى صورة جدول مبسط كالتالى:

جدول ٧-١ بيانات درجة الحرارة مع الزمن

الزمن	1	1.5	2	2.5	3	3.5	4	4.5	5
درجة الحرارة	150	160	100	130	80	100	125	105	80

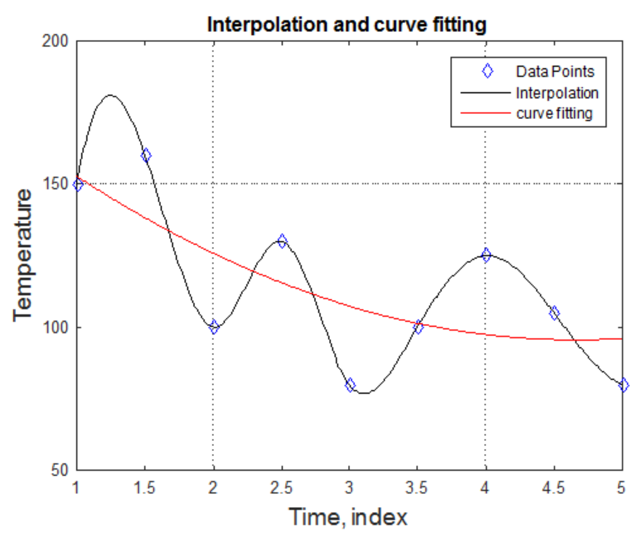
المطلوب يكون فى العادة هو إيجاد علاقة بين درجة الحرارة والزمن بحيث يمكن تقدير الحرارة عند أزمنة لم يتم قياس درجة الحرارة عندها، مثلا درجة الحرارة عند الزمن 2.27 كم تبلغ قيمتها بالضبط. قد تكون هناك حاجة مثلا لإيجاد علاقة فى صورة كثيرة حدود أو صورة من صور الدوال حتى يمكن تكاملها لاستخدام هذا التكامل فى أى تطبيق من التطبيقات. إن هذا هو ما نسميه الاستيفاء interpolation، بمعنى استيفاء ما بين النقط المحددة لمعرفة معلومات وسلوك هذه العملية عند مواضع تكون العملية غير معرفة عندها، وهناك تعريف للاستيفاء من ويكيبيديا بأنه طريقة لإنشاء نقاط جديدة فى مدى متقطع من نقاط البيانات المعروفة. وببساطة فإن الاستيفاء هو رسم منحنى مستمر يمر بجميع نقط البيانات والتعبير عن هذا المنحنى بأى صورة تحليلية، وهذا هو ما سنراه بالتفصيل فى هذا الفصل حيث سندرس طرق مختلفة لإجراء هذا الاستيفاء.

فى الكثير من الأحيان لا يهمنا أن يمر منحنى الاستيفاء بجميع النقط المعطاة ولكننا نريد رسم منحنى تقريبى يمر بين هذه النقاط بحيث يكون الفرق (أو الخطأ) بين النقاط المعطاة وهذا المنحنى أقل ما يمكن، وهذا هو ما نسميه تقريب المنحنيات، وهو مطلوب أيضا فى الكثير من التطبيقات، وسنقدم فى هذا الفصل طرقا مختلفة لرسم مثل هذه المنحنيات الملائمة أو التقريبية.

شكل ٧-١ يبين جدول درجة الحرارة السابق وقد تم رسمه فى صورة نقاط توضح مواضع النقاط المحددة فى الجدول ثم تم عمل استيفاء لهذه النقاط فى صورة منحنى متعدد القطع (سبلاين) spline باللون الأسود وهى أحد طرق الاستيفاء

التى سندرسها حيث نلاحظ أن منحنى الاستيفاء يمر بجميع النقاط المحددة فى الجدول. يبين الشكل أيضا منحنى تقريبى لهذه النقاط باللون الأحمر حيث نلاحظ كما ذكرنا أن هذا المنحنى لا يمر بجميع النقاط ولكنه يمر بينها بحيث يكون الفرق بين نقاطه والنقط المعطاه أقل ما يمكن. نقدم هنا أوامر ماتلاب المستخدمة فى هذا الرسم كتذكرة لأوامر الرسم أما أوامر الاستيفاء وتقريب المنحنى فسيتم شرحها بالتفصيل.

```
% difference between interpolation and curve fitting
x = 1:0.5:5;
y = [150 160 100 130 80 100 125 105 80];
x_new = 0:.01:5;
y_new = interp1(x,y,x_new,'spline');
plot(x,y,'bd')
hold
plot(x_new,y_new,'k', 'linewidth',1);
hold
c=polyfit(x,y,2);
yvals=polyval(c,x_new);
hold
plot(x_new,yvals,'r','linewidth',1);
title('Interpolation and curve fitting','fontsize',12)
ylabel('Temperature','fontsize',14)
xlabel('Time, index','fontsize',14)
legend('Data Points','Interpolation', 'curve fitting')
axis([1 5 50 200]);
hold; grid;
```



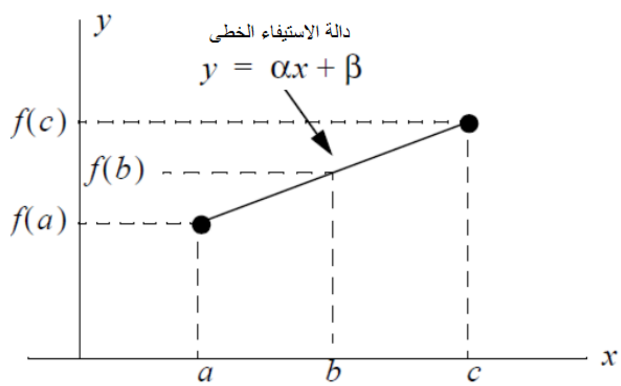
شكل ٧-١ الفرق بين الاستيفاء والمنحنى التقريبى لجدول درجات الحرارة

فوائد الاستيفاء أو تقريب المنحنى عديدة بجانب فائدة إمكانية تحديد الدالة عند نقاط لم تكن محددة ضمن مجموعة النقاط الأصلية فإننا سنرى استخدامها فى التكامل العددى وفى الحل العددى للمعادلات التفاضلية فى الفصول القادمة.

كما أنه تخيل أن لديك عدد كبير من نقاط البيانات المخزنة فى الحاسب فإنه بتحويل هذه البيانات إلى معادلة أو كثيرة حدود تمثلها بأحد طرق تقريب المنحنيات أو الاستيفاء فإن تخزين هذه البيانات سيشغل بالطبع مساحة أقل فى الذاكرة. طرق استيفاء البيانات عديدة وفى الغالب يتم تقسيمها على حسب طريقة التوصيل بين هذه النقاط. فمثلا إذا كان التوصيل بين النقاط يتم عن طريق خطوط مستقيمة فإننا نسميه الاستيفاء الخطى، وإذا كان عن طريق كثيرة حدود فإنه يسمى استيفاء كثيرة الحدود. هناك أيضا الاستيفاء عن طريق استخدام دوال أساسية مثل طريقة لاجرانج Lagrange ونيوتين Newton وكل هذه الطرق سيتم شرحها بالتفصيل.

٧-١ الاستيفاء الخطى Linear interpolation

الاستيفاء الخطى هو أبسط أنواع الاستيفاء حيث يتم التوصيل بين كل نقطتين متتاليتين بخط مستقيم أو معادلة من الدرجة الأولى، وأى نقطة أخرى بين هاتين النقطتين يمكن الحصول عليها بالتعويض فى معادلة هذا الخط المستقيم. إفترض أن لدينا النقطة الأولى التى إحداثياتها هى (x=a ، y=f(a))، والنقطة الثانية التى إحداثياتها هى (x=c، y=f(c)) ونريد التوصيل بينهما بخط مستقيم بحيث منه نحسب قيمة الدالة عند النقطة x=b والمفترض أن تكون y=f(b). شكل ٧-٢ يبين هاتين النقطتين والخط الواصل بينهما. وأما معادلة هذا الخط فيمكن كتابتها كما يلى:



شكل ٧-٢ الاستيفاء الخطى أو الاستيفاء من الدرجة الأولى

$$y = f(x) = \alpha x + \beta \qquad (٧-١)$$

حيث α و β ثابتان يمكن تحديدهما بالتعويض فى المعادلة (٧-١) بالنقطتين (a, f(a)) و (c, f(c)) فنحصل على معادلتين يمكن حلهما لتحديد كل من α و β وفى النهاية يمكن كتابة معادلة الخط المستقيم كما يلى:

$$y = f(a) + \frac{x-a}{c-a}[f(c) - f(a)] \qquad (٧-٢)$$

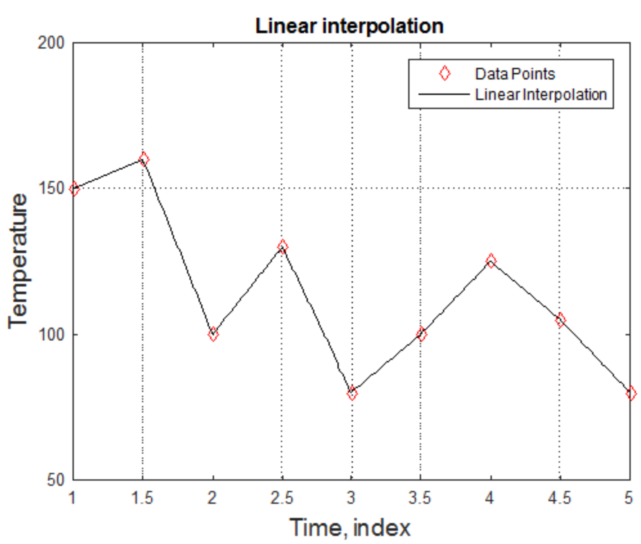
لاحظ أنه بالتعويض فى هذه المعادلة عن x=a نحصل على y=f(a) وبالتعويض عن x=c نحصل على y=f(c)، وبوضع x=b نحصل على $y = f(b) = f(a) + \frac{b-a}{c-a}[f(c) - f(a)]$ حيث b تعتبر نقطة استيفاء لم تكن معرفة على الخط المستقيم.

الآن ماذا عن باقى النقاط التى فى مجموعة البيانات مثل التى فى جدول ٧-١ مثلا؟ الذى يحدث هو أن نقاط هذا الجدول يتم أخذها على التوالى كل نقطتين مع بعضهما والتوصيل بينهما بخط مستقيم لنحصل على الاستيفاء الخطى أو الاستيفاء من الدرجة الأولى لنقاط البيانات التى فى جدول ٧-١. يقدم برنامج ماتلاب الدالة interp1(x,y,'linear') لعمل استيفاء لمجموعة النقاط المحددة بالمتجهين x و y واللذان يجب أن يكون لهما نفس الطول. المعامل 'linear' أو "خطى" فى هذه الدالة يعنى أن الاستيفاء المطلوب هو من النوع الخطى. لاحظ أن نفس هذه الدالة تم استخدامها لعمل الاستيفاء لنفس مجموعة النقاط ولكن الاستيفاء هناك من النوع سبلاين الذى سيتم شرحه فيما بعد. شكل ٧-٣ يوضح نتيجة الاستيفاء الخطى لهذه البيانات التى فى جدول ٧-١ والبرنامج المستخدم فى ذلك هو كالتالى:

```
1- % Linear interpolation
2- x = 1:0.5:5;
3- y = [150 160 100 130 80 100 125 105 80];
4- x_new = 0:.01:5;
5- y_new = interp1(x,y,x_new,'linear');
6- plot(x,y,'rd'); hold;
7- plot(x_new,y_new,'k', 'linewidth',1);
8- hold; grid;
9- title('Linear interpolation','fontsize',12)
10- ylabel('Temperature','fontsize',14)
11- xlabel('Time, index','fontsize',14)
12- legend('Data Points','Linear Interpolation')
13- axis([1 5 50 200]);
```

نكرر هنا أن x و y هى نقاط البيانات المراد استيفاؤها وأما x_new فهى نقاط رسم منحنى الاستيفاء وقد تم تحديدها فى الأمر رقم ٤ فى البرنامج حيث x_new ستتغير من 0 حتى 5 أو أى قيمة اختيارية وبخطوة اختيارية أيضا مقدارها 0.01 وسيقوم أمر الاستيفاء بحساب y_new عند كل قيمة من قيم x_new ويتم رسم المنحنى بالكامل بأمر الرسم plot فى السطر رقم ٧.

بعد انتهاء الاستيفاء يمكنك الآن استخدام نفس دالة الاستيفاء لحساب درجة الحرارة عند أى زمن غير موجود فى بيانات الجدول ٧-١. فمثلا لحساب



شكل ٧-٣ الاستيفاء الخطى لبيانات الجدول ٧-١

درجة الحرارة عند الزمن 4.2 نكتب الأمرين التاليين فى نهاية البرنامج السابق لإدخال الزمن:

```
a = input(['Enter any time value you want to interpolate a=']); %e.g [50]
y_new = interp1(x,y,a,'linear');
disp(y_new)
```

والذى كانت نتيجة تنفيذه كالتالى:

Enter any time value you want to interpolate a= 4.2
117

قارن هذه النتيجة مع منحنى الاستيفاء الذى فى شكل ٧-٣.

٧-٢ استيفاء كثيرات الحدود Polynomial interpolation

حتى يكون كلامنا أكثر عمومية فإننا سنفترض عدد n من نقاط البيانات (x_i, y_i) حيث i تتغير من 1 إلى n. المطلوب هو البحث عن كثيرة حدود من الدرجة n تمر بكل هذه النقاط مع ملاحظة أنه لا توجد نقطتين (x_i, y_i) و (x_j, y_j) وتكون لهما i=j لأن ذلك يعنى خطا رأسيا عند المركبة x_i مما يعنى عدم استمرارية أو عدم اتصالا ولا يمكن تحديد كثيرة حدود تمر عند هذا الانكسار، مما يعنى x_i يجب أن تكون مميزة أو متفردة أو غير متكررة. جدول ٧-٢ يبين هذه النقاط.

جدول ٧-٢ نقاط البيانات المراد استفاؤها بكثيرة حدود من الدرجة n.

x_0	x_1	x_2	... ...	... .	x_n
y_0	y_1	y_2	....	....	y_n

سنفترض أن كثيرة الحدود من الدرجة n التى تمر بهذا العدد n من النقاط ستكون على الصورة العامة التالية:

$$p_n(x) = 1 + a_1x^1 + a_2x^2 + a_3x^3 + \cdots + a_nx^n \qquad (٧-٣)$$

حيث a_i و i تتغير من 1 حتى n هى معاملات كثيرة الحدود المجهولة والمطلوب تحديدها. بالتعويض بنقاط البيانات n التى فى جدول ٧-٢ فى المعادلة (٧- ٣) نحصل على عدد n من المعادلات فى عدد n من المجاهيل، a_n، التى يمكن حلها لإيجاد هذه المجاهيل كما يلى:

$$\begin{aligned} y_0 &= 1 + a_1x_0^1 + a_2x_0^2 + a_3x_0^3 + \cdots + a_nx_0^n \\ y_1 &= 1 + a_1x_1^1 + a_2x_1^2 + a_3x_1^3 + \cdots + a_nx_1^n \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ y_n &= 1 + a_1x_n^1 + a_2x_n^2 + a_3x_n^3 + \cdots + a_nx_n^n \end{aligned} \qquad (٧-٤)$$

هذه المعادلات يمكن وضعها فى صورة مصفوفات كالتالى:

$$\begin{pmatrix} y_0 \\ y_1 \\ \dots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_0^1 & x_0^2 & \dots & x_0^n \\ 1 & x_1^1 & x_1^2 & \dots & x_1^n \\ 1 & \dots & \dots & \dots & \dots \\ 1 & x_n^1 & x_n^2 & \dots. & x_n^n \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \dots. \\ a_n \end{pmatrix} \qquad (٥-٧)$$

فى المعادلة (٥–٥) متجه المجاهيل المطلوب تحديد قيمتها هو المتجه a ومتجه الثوابت هو المتجه y المعطى فى جدول النقاط ومصفوفة المعاملات هى المصفوفة x وهى تسمى مصفوفة فاندرموند Vandermonde وأحيانا يرمز لها بالرمز V نسبة للعالم ألكسندر فاندرموند، وهذه المصفوفة لها مميزات خاصة فى حساب محددتها يمكن الاطلاع عليها فى أى كتاب فى الجبر الخطى، كما أن الماتلاب يعطى هذه المصفوفة فورا بمجرد إعطاؤه المتجه $x=(x_0,x_1, \dots x_n)$ كما يلى:

```
>> x=[1 2 3 4];
>> v=vander(x)
v =
    1    1    1    1
    8    4    2    1
   27    9    3    1
   64   16    4    1
```

حيث بإعطاء المتجه x=[1 2 3 4] كانت المصفوفة v تحتوى فى صفها الأول أسس العنصر الأول، 1، فى المتجه x، وتحتوى فى صفها الثانى أسس العنصر الثانى، 2، فى المتجه x، وفى الصف الثالث أسس العنصر الثالث، 3، وهكذا.

يمكن قلب أعمدة هذه المصفوفة باستخدام الأمر التالى:

```
>> fliplr(v)
ans =
    1    1    1    1
    1    2    4    8
    1    3    9   27
    1    4   16   64
```

المعادلة (٥–٧) كما ذكرنا عبارة عن عدد n من المعادلات فى عدد n من المجاهيل وهى عناصر المصفوفة a ويمكن استخدام أى طريقة من طرق حل نظم المعادلات الخطية التى درسناها فى الفصل الثالث (جاوس لحذف المجاهيل)، أو الطرق التكرارية التى درسناها فى الفصل الرابع، أو طريقة القسمة العكسية للمصفوفات a=V\y. لاحظ أن نظام المعادلات السابق هو نظام خطى فى متجه المجاهيل a، وهذا هو السبب فى إمكانية حله باستخدام طرق الفصلين ٣ و ٤.

مثال ٧–١

أوجد كثيرة الحدود من الدرجة الثالثة التى تمر بالنقاط الأربع التالية: (10 ,2-) و (4 ,1-) و (6 ,1) و (2,3). تذكر أن ترتيب النقاط ليس له أى تأثير على الحل. نظام المعادلات يمكن كتابته كالتالى:

$$
\begin{aligned}
10 &= 1 + a_1(-2)^1 + a_2(-2)^2 + a_3(-2)^3 \\
4 &= 1 + a_1(-1)^1 + a_2(-1)^2 + a_3(-1)^3 \\
6 &= 1 + a_1(1)^1 + a_2(1)^2 + a_3(1)^3 \\
3 &= 1 + a_1(2)^1 + a_2(2)^2 + a_3(2)^3
\end{aligned}
$$

والذى يمكن كتابته فى صورة المصفوفات كما يلى:

$$
\begin{pmatrix} 1 & -2 & 4 & -8 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \end{pmatrix}
\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix}
=
\begin{pmatrix} 10 \\ 4 \\ 6 \\ 3 \end{pmatrix}
$$

وهذا النظام يمكن حله بأى طريقة من طرق الحل التى نعرفها، ولكن باستخدام ماتلاب حصلنا على الحل التالى:

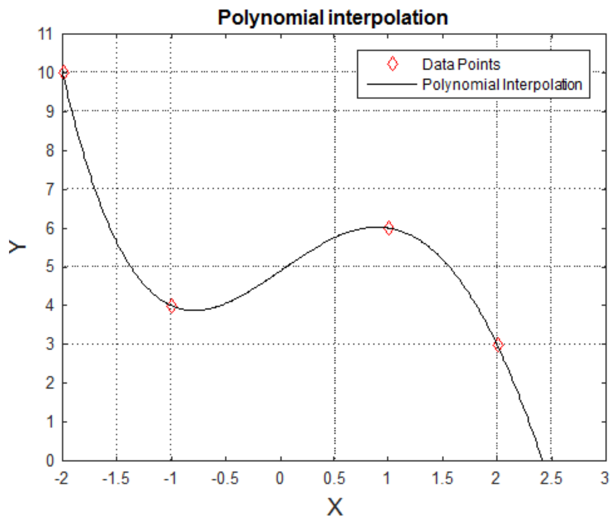
```
>> V=[1 -2 4 -8;1 -1 1 -1;1 1 1 1;1 2 4 8];
>> y=[10;4;6;3];
>> a=V\y
a =
   4.5000
   1.9167
   0.5000
  -0.9167
```

يستخدم ماتلاب الأمر a=polyfit(x,y,n) حيث x و y هما إحداثيات النقط المطلوب استيفاؤها و n هى عدد هذه النقاط أو بمعنى آخر درجة كثيرة الحدود الناتجة، و a هى معاملات كثيرة الحدود تعطى بترتيب تصاعدى بمعنى أن أول عنصر فى المتجه a يكون معامل x_0. الدالة y=polyval(a,x) تعطى القيمة y لكثيرة الحدود التى معاملاتها a والتى تم حسابها عن طريق الأمر polyfit() عند الإحداثى x.

البرنامج التالى سيرسم كثيرة الحدود الناتجة من استيفاء النقاط الأربعة التى فى مثال ٧–١ ويحسب قيمة هذه الدالة عند النقاط x=-2 حتى x=3 وبخطوة مقدارها 0.01 باستخدام الدالة y_new=polyval(a,x_new) كما فى البرنامج. شكل الاستيفاء الناتج موضح فى شكل ٧–٤.

```
1-  % polynomial interpolation
2- x = [-2 -1 1 2];
3- y = [10 4 6 3];
4- x_new = -2:.01:3;
```

```
5- a= polyfit(x,y,4);
6- y_new=polyval(a,x_new);
7- plot(x,y,'rd'); hold;
8- plot(x_new,y_new,'k',
'linewidth',1);
9- hold; grid;
10-  title('Polynomial
interpolation','fontsize',12)
11-  ylabel('Y','fontsize',14)
12-  xlabel('X','fontsize',14)
13-  legend('Data
Points','Polynomial Interpolation')
14-  axis([-2 3 0 11]);
```

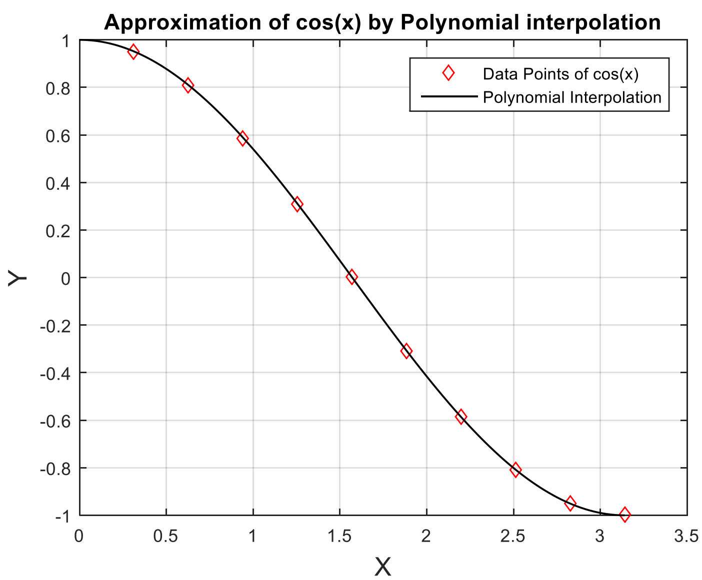


شكل ٧–٤ استيفاء كثيرة الحدود لنقاط المثال ٧–١

يمكن استخدام الاستيفاء بكثيرة الحدود أو أى طريقة من طرق الاستيفاء لتقريب الدوال المعقدة. فمثلا إذا كان لديك أى دالة مركبة وتريد رسمها فى مدى معين فإنه يمكنك أخذ أى عدد من النقاط على هذه الدالة ثم عمل استيفاء لهذه النقاط بكثيرة حدود درجتها تساوى عدد هذه النقاط، والمثال التالى يبين ذلك.

مثال ٧–٢

استخدم كثيرة حدود من الدرجة العاشرة لتقريب الدالة f(x)=cos(x) فى المدى من 0 حتى π. الدالة f(x) يمكن أن تكون أى دالة بأى درجة من التعقيد. الفكرة هنا هى أننا سنحسب ١٠ نقاط للدالة f(x) فى المدى من 0 حتى π ثم نقوم بإيجاد كثيرة الحدود من الدرجة العاشرة التى تستوفى هذه النقاط العشرة. شكل ٧–٥ يبين النقاط العشرة على الدالة f(x)=cos(x) وقد تم استيفاؤها بكثيرة الحدود. والبرنامج التالى يوضح ذلك:



شكل ٧–٥ تقريب الدالة f(x)=cos(x) بكثيرة حدود من الدرجة العاشرة

```
1- % Using polynomial interpolation to approximate functions
2- i=linspace(1,10,10);
```

```
x = i*pi/10;
y = cos(x);
plot(x,y,'rd'); hold;
x_new = 0:.01:pi;
a= polyfit(x,y,10);
y_new=polyval(a,x_new);
plot(x_new,y_new,'k', 'linewidth',1);
hold; grid;
title('Approximation of cos(x) by Polynomial
 interpolation','fontsize',12)
ylabel('Y','fontsize',14)
xlabel('X','fontsize',14)
legend('Data Points of cos(x)','Polynomial Interpolation')
```

يجب أن نتذكر هنا أن مصفوفة فاندرموند يجب أن تكون غير متفردة (محددها لا تساوى الصفر)، ويمكن الاثبات رياضيا أنه يكفى ألا تكون هناك نقطتين فى مجموعة النقاط لهما نفس المركبة x، أى لا يكون هناك خطا رأسيا فى مجموعة البيانات أو نقطتين فوق بعض رأسيا تماما. يجب أن نتذكر أيضا أن الاستيفاء بمصفوفة فاندرموند يكون مكلف حسابيا، بمعنى أنه يحتاج لوقت طويل للحساب بالذات إذا كانت أبعاد المصفوفة كبيرة أو إذا كان عد د نقاط البيانات كبيرا.

٧-٣ خواريزم هورنر Horner

هورنر هو عالم رياضيات انجليزى وضع طريقة يتم بها حساب أى كثيرة حدود عند أى قيمة معينة للمتغير المستقل تكون أسرع بكثير عن التعويض المباشر فى كثيرة الحدود ولكى نفهم ذلك نبينه بالمثال التالى:

مثال ٧-٣

مطلوب حساب قيمة كثيرة الحدود التالية عند x=2:

$$p(x)=4x^5-3x^4+7x^3+6x^2+3x+9$$

الطريقة الأولى هى التعويض المباشر عن x=2 فى المعادلة السابقة فنحصل على:

$$p(x)=4(2)^5-3(2)^4+7(2)^3+6(2)^2+3(2)+9=175$$

الطريقة الثانية هى أن نضع كثيرة الحدود السابقة فى الصورة التالية قبل التعويض:

$$p(x)=\left(\left(\left((4x-3)x+7\right)x+6\right)x+3\right)x+9$$

وبالتعويض عن x=2 فى هذه الصورة لكثيرة الحدود نحصل على:

$$p(x) = \left(\left(\left((4(2) - 3)(2) + 7\right)(2) + 6\right)(2) + 3\right)(2) + 9 = 175$$

ما هو الفرق فى الطريقتين السابقتين لحساب كثيرة الحدود عند x=2؟ الفرق كبير، ففى الطريقة الأولى بالنظر إلى كثيرة الحدود سنجد أن الحاسب سيحتاج إلى ٥ عمليات ضرب ليحسب x^5 ثم عملية ضرب إضافية ليضرب فى ٤ وذلك للمقدار الأول $4x^5$. فى المقدار الثانى، $3x^4$، سيحتاج الحاسب إلى ٤ عمليات ضرب ليحسب x^4 ثم عملية أخرى ليضرب فى ٣. فى المقدار الثالث، $7x^3$، سيحتاج الحاسب إلى ٣ عمليات ضرب ليحسب x^3 ثم عملية أخرى ليضرب فى 7. فى المقدار الرابع، $6x^2$، سيحتاج الحاسب إلى عمليتى ضرب ليحسب x^2 ثم عملية أخرى ليضرب فى 6. فى المقدار الخامس سيحتاج الحاسب لعملية ضرب واحدة ليضرب x فى ٣، ثم فى النهاية يجمع ٩. بذلك يكون مجموع عمليات الضرب فقط وهو ما يهمنا هنا هو ١٩ عملية ضرب لحساب قيمة كثيرة الحدود بالتعويض المباشر فى صورتها الأولى. فى الصورة الثانية لكثيرة الحدود سيحتاج الحاسب إلى ٥ عمليات ضرب فقط، وهذا يعتبر تخفيضا كبيرا فى عمليات الضرب التى تكون أكثر العمليات الحسابية استهلاكا للوقت. لذلك فإن ماتلاب والكثير من التطبيقات تستخدم هذه الطريقة عند حساب أى كثيرة حدود.وعلى ذلك يمكن كتابة الصورة العامة لأى كثيرة حدود بطريقة هورنر كما يلى:

$$p(x) = \left(\left(\left((a_n x - a_{n-1})x + a_{n-2}\right)x + \cdots + a_2\right)x + a_1\right)x + a_0 \qquad (٦-٧)$$

٧-٤ طريقة لاجرانج للاستيفاء

لقد سميت هذه الطريقة على إسم لاجرانج Lagrange وهو عالم رياضيات فرنسى. تقوم فكرة طريقة لاجرانج على تكوين كثيرة الحدود من مجموعة من المقادير التى كل منها يساوى واحد عند أحد النقاط ويساوى صفر عند باقى النقاط، وهذه المقادير تسمى الدوال الأساسية. ولذلك بفرض أن لدينا عدد n+1 من النقاط، (x_0,y_0) و (x_1,y_1) و (x_2,y_2) و.... حتى (x_n,y_n)، فإنه يمكن كتابة كثيرة الحدود بطريقة لاجرانج على الصورة التالية:

$$L(x) = \sum_{i=0}^{n} y_i r_i(x) \qquad (٧-٧)$$

حيث r_i هى أحد الدوال الأساسية. أى أنه بالتعويض بمجموعة النقاط السابقة فى المعادلة (٧-٧ يمكن إعادة كتابتها كما يلى:

$$L(x) = y_0 r_0(x) + y_1 r_1(x) + y_2 r_2(x) + \cdots + y_n r_n(x) \qquad (٨-٧)$$

حيث y_0 و y_1 و... حتى y_n هى المركبة الرأسية لكل نقطة من النقط المعطاة، و $r_0(x)$ عبارة عن دالة أساس فى المتغير x تساوى 1 عندما $x=x_0$ وتساوى صفر عند أى قيمة أخرى للمتغير x، و $r_1(x)$ عبارة دالة أساس فى المتغير x

تساوى 1 عندما $x=x_1$ وتساوى صفر عند أى قيمة أخرى للمتغير x، وهكذا باقى دوال الأساس حتى الدالة $r_n(x)$ التى تساوى 1 عندما $x=x_n$ وتساوى صفر عند أى قيمة أخرى للمتغير x. نتيجة ذلك كله أن كثيرة الحدود $L(x)$ ستمر بجميع النقاط المعطاة، وما يبقى فقط هو معرفة أو تحديد دوال الأساس $r_i(x)$، وهذا هو ما فعله لاجرانج أيضا حيث حدد هذه الدوال كما يلى:

$$r_0(x) = \frac{(x-x_1)(x-x_2).....(x-x_n)}{(x_0-x_1)(x_0-x_2)......(x_0-x_n)} \qquad (٧-٩)$$

بوضع $x=x_0$ فى المعادلة (٧- ٩) سنجد أن $r_0(x_0)=1$، وبوضع $x=x_1$ أو $x=x_2$ وهكذا حتى $x=x_n$ فإن $r_0(x)$ ستكون صفر فى كل هذه الأحوال. الدالة $r_0(x)$ تسمى دالة أساس لاجرانج من الرتبة صفر لعدد n-1 من النقاط، وبنفس الطريقة يمكن كتابة $r_1(x)$ و $r_2(x)$ حتى $r_n(x)$، وفى النهاية يمكن إعادة كتابة كثيرة حدود لاجرانج التى فى المعادلة (٧- ٨) كما يلى:

$$L(x) = y_0 \frac{(x-x_1)(x-x_2)...(x-x_n)}{(x_0-x_1)(x_0-x_2)......(x_0-x_n)} + y_1 \frac{(x-x_0)(x-x_2)...(x-x_n)}{(x_1-x_0)(x_1-x_2)......(x_1-x_n)} + \cdots + y_n \frac{(x-x_0)(x-x_1)...(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)......(x_n-x_{n-1})} \qquad (٧-١٠)$$

لاحظ أن دالة أساس لاجرانج $r_i(x)$ تساوى حاصل ضرب جميع الأقواس الخطية فيما عدا القوس $(x-x_i)$ مقسوما على حاصل ضرب جميع هذه الأقواس الخطية عند $x=x_i$، فيما عدا عند $x=x_i$. يمكن كتابة دوال أساس لاجرانج فى صورة مدمجة كما يلى:

$$L(x) = y_0 \prod_{\substack{j=0 \\ j\neq 0}}^{n} \frac{(x-x_j)}{(x_0-x_j)} + y_1 \prod_{\substack{j=0 \\ j\neq 1}}^{n} \frac{(x-x_j)}{(x_1-x_j)} + \cdots + y_n \prod_{\substack{j=0 \\ j\neq n}}^{n} \frac{(x-x_j)}{(x_n-x_j)} \qquad (٧-١١)$$

ويمكن كتابة الصورة المدمجة الكلية لكثيرة حدود لاجرانج التى فى المعادلة (٧- ٧) كما يلى:

$$L(x) = \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j\neq 0}}^{n} \frac{(x-x_j)}{(x_i-x_j)} \qquad (٧-١٢)$$

مثال ٧- ٤

استخدم طريقة لاجرانج لإيجاد دالة استيفاء تمر بالنقاط الأربع التالية: (2, 10-) و (1-, 4) و (1, 6) و (2,3).

تذكر أن ترتيب النقاط ليس له أى تأثير على الحل أيضا. كثيرة حدود لاجرانج لأربع نقاط يمكن كتابتها كما يلى:

$$L(x) = y_0 \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1 \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} + y_2 \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

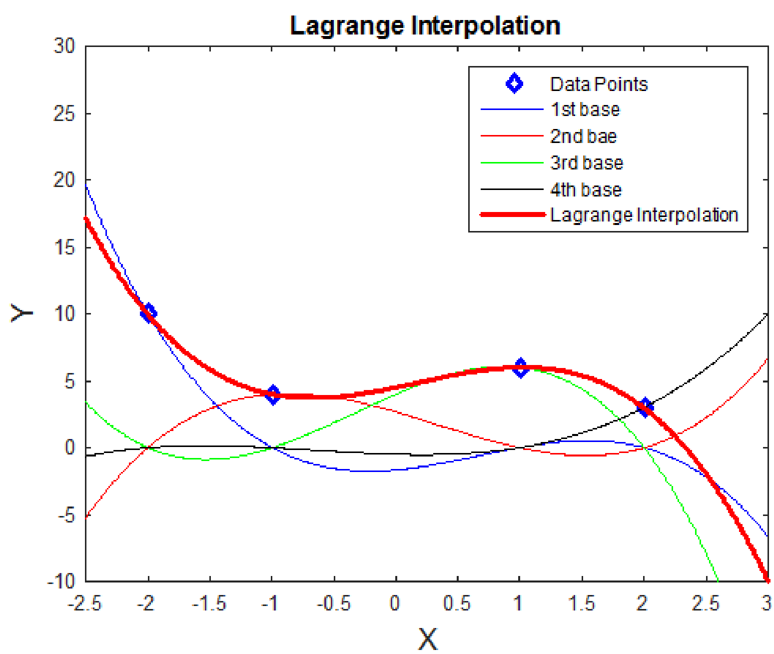
$$L(x) = (10)\frac{(x-(-1))(x-1)(x-2)}{(-2-(-1))(-2-1)(-2-2)} + (4)\frac{(x-(-2))(x-1)(x-2)}{(-1-(-2))(-1-1)(-1-2)} + (6)\frac{(x-(-2))(x-(-1))(x-2)}{(1-(-2))(1-(-1))(1-2)} + (3)\frac{(x-(-2))(x-(-1))(x-1)}{(2-(-2))(2-(-1))(2-1)}$$

والتى يمكن وضعها على الصورة:

$$L(x) = (10)\frac{(x+1)(x-1)(x-2)}{(-1)(-3)(-4)} + (4)\frac{(x+2)(x-1)(x-2)}{(1)(-2)(-3)} + (6)\frac{(x+2)(x+1)(x-2)}{(3)(2)(-1)} + (3)\frac{(x+2)(x+1)(x-1)}{(4)(3)(1)}$$

وهكذا يمكن الاستمرار فى الاختصارات حتى الوصول إلى صورة نهائية لكثيرة الحدود.

البرنامج التالى يقوم بتنفيذ هذا الاستيفاء على النقاط الأربعة مع بيان دوال الأساس الأربعة أيضا وشكل ٧-٦ يبين نتيجة تنفيذ هذا البرنامج. لاحظ أن كل دالة أساس مقابلة لأى نقطة تساوى المركبة y لهذه النقطة وتساوى صفر عند باقى النقط. مثلا دالة الأساس المقابلة للنقطة (2,10-) والمرسومة بالمنحنى الأزرق تساوى 10 عندما x=-2 وتساوى صفر عند النقاط الثلاث الأخرى، وهكذا لباقى دوال الأساس. مجموع كل دوال الأساس يعطى منحنى لاجرانج للاستيفاء كما فى المنحنى الأحمر السميك.



شكل ٧-٦ استيفاء النقاط الأربعة فى المثال ٧-٤ بطريقة لاجرانج مع بيان دوال الأساس الأربعة.

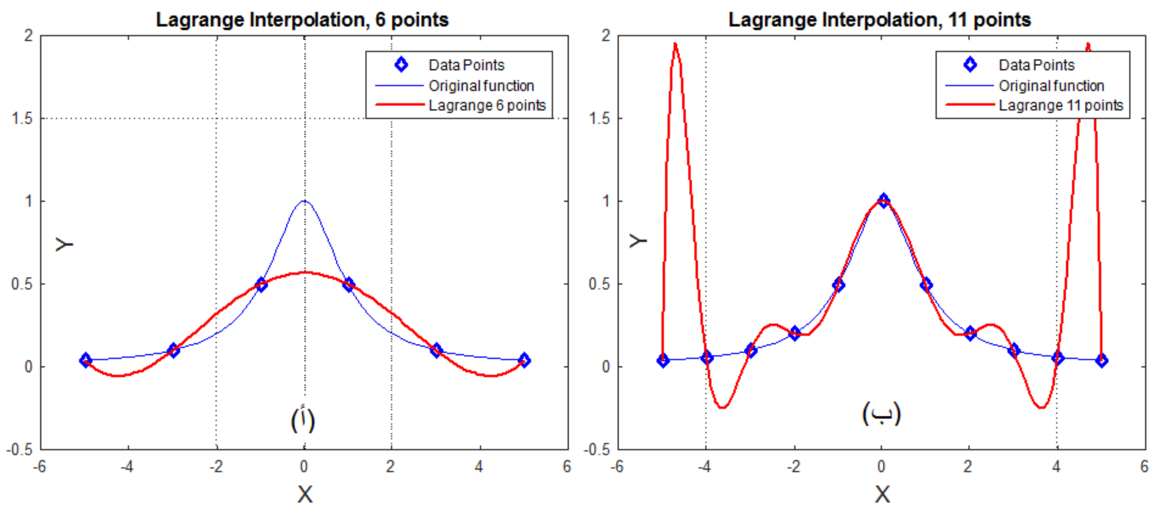
```
1- % Polynomial interpolation by Lagrange method
2- pointx = [-2 -1 1 2];
```

```
3- pointy = [10 4 6 3];
4- x_new = -2.5:.01:3;
5- plot(pointx,pointy,'bd','Linewidth',2); hold on;
6- m=size(pointx,2);
7- r=ones(m,size(x_new,2));
8- for i=1:4
9-    for j=1:4
10-      if (i~=j)
11-         r(i,:)=r(i,:).*(x_new-pointx(j))/(pointx(i)-pointx(j));
12-      end
13-   end
14- End
15- plot(x_new,pointy(1)*r(1,:),'b',x_new,pointy(2)*r(2,:),'r',x_new,point
    y(3)*r(3,:),'g',x_new,pointy(4)*r(4,:),'k');
16- hold on;
17- y_new=0;
18- for i=1:4
19-   y_new=y_new+pointy(i)*r(i,:);
20- end
21- plot(x_new,y_new,'r','Linewidth',2);
22- title('Lagrange Interpolation','fontsize',12)
23- ylabel('Y','fontsize',14)
24- xlabel('X','fontsize',14)
25- legend('Data Points','1st base','2nd bae','3rd base','4th
    base','Lagrange Interpolation');
26- axis([-2.5 3 -10 30]);
```

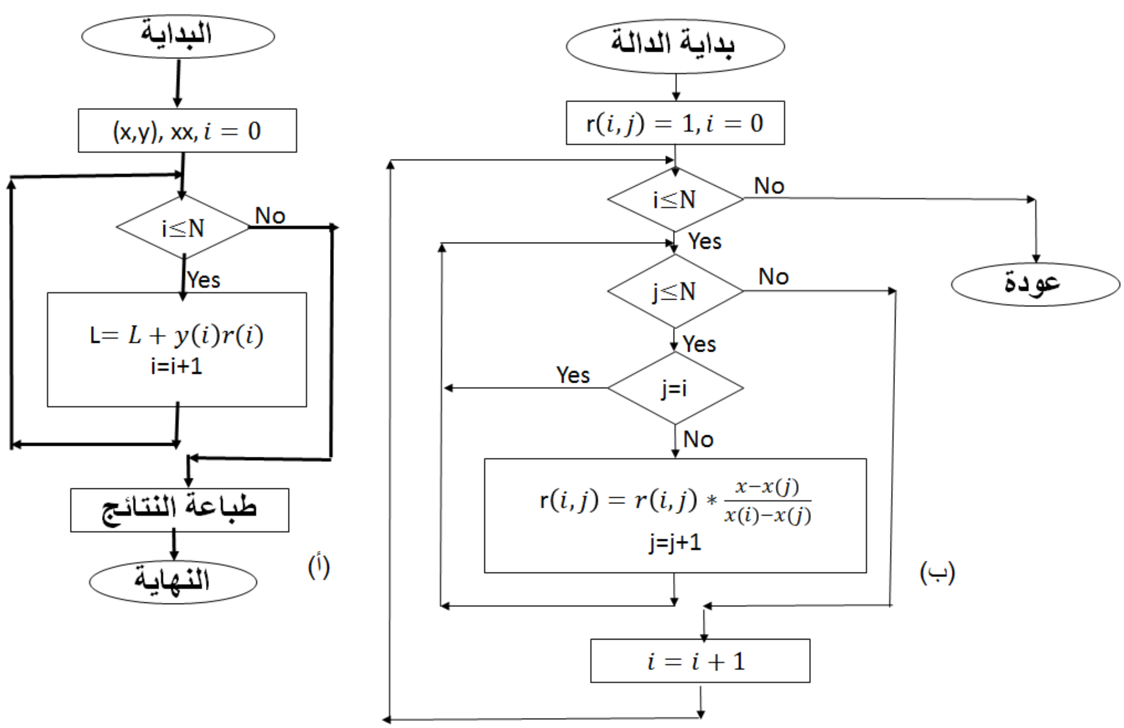
فى الأوامر السابقة pointx و pointy هما المركبتين x و y للنقاط المراد استيفاؤها، بينما x_new فهى المركبة x التى سيتم التعويض بها فى منحنى الاستيفاء الناتج أو دوال الأساس التى سيتم رسمها. الأوامر من ٨ إلى ١٤ عبارة عن حلقتين متداخلتين لحساب دوال الأساس r(i) عند جميع النقاط x_new. الأمر ١٥ يرسم دوال الأساس الأربعة كل منها بلون مختلف بعد إتمام حسابها والخروج من الحلقتين. بعد ذلك يتم الدخول فى حلقة فى الأوامر ١٨ حتى ٢٠ لحساب منحنى الاستيفاء عن طريق تجميع دوال الأساس الأربعة التى سبق حسابها، ثم يتم رسم دالة الاستيفاء النهائية فى الأمر ٢١ ثم كتابة عنوان الشكل وأسماء المحاور ومفتاح الشكل وحدود المحاور فى باقى الأوامر.

كما ذكرنا من قبل فإنه من الممكن استخدام الاستيفاء فى تقريب الدوال، وهذا ما تم فى شكل ٧-٧ حيث استخدمنا الاستيفاء بطريقة لاجرانج لتقريب الدالة $f(x)=\frac{1}{1+x^2}$، مرة باستخدام ٦ نقاط كما فى شكل ٧-٧(أ)، ومرة باستخدام ١١ نقطة كما فى شكل ٧-٧(ب). نلاحظ من هذا الشكل أن زيادة عدد نقاط الاستيفاء يحسن من دقة التقريب فى منتصف الشكل، ولكنه يضعف من هذه الدقة عند أطراف الشكل كما نلاحظ فى شكل ٧-٧(ب) حيث تزيد ذبذبات منحنى التقريب بدرجة كبيرة عند أطراف الشكل. سيكون هناك علاج لهذه المشكلة مع استخدام استيفاء مقاطع الاسبلاين الذى سنقدمه فى نهاية هذا الفصل.

شكل ٧–٨ (أ و ب) يبين خريطة تدفق للبرنامج الأساسى وملف دالة يتم النداء عليه من البرنامج الأساسى لحساب دوال لاجرانج الأساسية بعد إدخال النقاط المراد استيفاؤها وعددها إلى البرنامج الأساسى. لقد تركنا كتابة هذا البرنامج كتمرين يقوم القارىء بتنفيذه واختباره.



شكل ٧–٧ استخدام الاستيفاء لتقريب الدوال (أ) باستخدام ٦ نقاط، و (ب) باستخدام ١١ نقطة



شكل ٧–٨ مخطط التدفق للبرنامج الأساسى وملف دالة لحساب استيفاء لاجرانج (أ) البرنامج الأساسى (ب) ملف الدالة

٧-٥ طريقة نيوتن للاستيفاء

بنفس الطريقة سنفترض أن لدينا عدد n+1 من النقاط (x_0,y_0) و (x_1,y_1) و .... حتى (x_n,y_n) والمطلوب هو استيفاء هذه النقاط عن طريق كثيرة حدود من الدرجة n تمر بجميع هذه النقاط، ولقد رأينا ذلك فى طريقة لاجرانج وطريقة فاندرموند. فى طريقة نيوتن يتم وضع هذه الكثيرة حدود أو الدالة كالتالى:

$$P(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) + \cdots + a_n(x - x_0)(x - x_1) \dots (x - x_{n-1}) \quad (١٣-٧)$$

فى المعادلة (٧- ١٣) بوضع $x=x_0$ سنحصل على:

$$P(x_0)=y_0=a_0 \quad (١٤-٧)$$

وبوضع $x=x_1$ نحصل على:

$$P(x_1)=y_1=y_0+a_1(x_1-x_0) \quad (١٥-٧)$$

ومنها نحصل على a_1 كما يلى:

$$a_1 = \frac{y_1-y_0}{x_1-x_0} = f(x_0.x_1) \quad (١٦-٧)$$

حيث $f(x_0,x_1)$ هى تسمية اختيارة لقسمة الفروق $\frac{y_1-y_0}{x_1-x_0}$ ولذلك فهى تسمى قسمة الفروق divided difference من الدرجة الأولى.

بوضع $x=x_2$ فى المعادلة (٧- ١٣) نحصل على:

$$P(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) \quad (١٧-٧)$$

وبمعلومية a_0 من المعادلة (٧- ١٤) و a_1 من المعادلة (٧ -١٦) يمكننا حساب a_2 كما يلى:

$$a_2 = \frac{\left[\left({(y_2-y_1)}/{(x_2-x_1)}\right)-\left({(y_1-y_0)}/{(x_1-x_0)}\right)\right]}{(x_2-x_0)} = f(x_0.x_1.x_2) \quad (١٨-٧)$$

حيث $f(x_0,x_1,x_2)$ هى تسمية اختيارة لقسمة الفروق الموضح فى المعادلة (٧- ١٨) ولذلك فهى تسمى قسمة الفروق divided difference من الدرجة الثانية. لاحظ أن قسمة الفروق من الدرجة الثانية يمكن كتابتها بدلالة قسمة الفروق من الدرجة الأولى كما يلى:

$$f(x_0.x_1.x_2) = \frac{f(x_1.x_2)-f(x_0.x_1)}{(x_2-x_0)} \quad (١٩-٧)$$

بنفس الطريقة يمكن الاستمرار فى حساب a3 التى ستكون عبارة عن قسمة فروق من الدرجة الثالثة والتى يمكن كتابتها بدلالة قسمة الفروق من الدرجة الثانية كما يلى:

$$a_3 = f(x_0.x_1.x_2.x_3) = \frac{f(x_1.x_2.x_3)-f(x_0.x_1.x_2)}{(x_3-x_0)} \quad (٢٠-٧)$$

فى النهاية يمكن كتابة an كقسمة فروق من الدرجة n والتى يمكن كتابتها بدلالة قسمة الفروق من الدرجة n-1 كما يلى:

$$a_n = f(x_0.x_1.\dots.x_n) = \frac{f(x_1.x_2\dots x_n)-f(x_0.x_1\dots x_{n-1})}{(x_n-x_0)} \quad (٢١-٧)$$

وعلى ذلك يمكن كتابة المعادلة (٧- ١٣) بدلالة الفروق المقسومة كما يلى:

$$P(x) = f(x_0) + f(x_0.x_1)(x-x_0) + f(x_0.x_1.x_2)(x-x_0)(x-x_1) + f(x_0.x_1.x_2.x_3)(x-x_0)(x-x_1)(x-x_2) + \cdots + f(x_0.x_1\dots.x_n)(x-x_0)(x-x_1)\dots(x-x_{n-1}) \quad (٢٢-٧)$$

والتى يمكن كتابتها فى الصورة المدمجة التالية:

$$P(x) = f(x_0) + \sum_{k=1}^{n} f(x_0.x_1\dots.x_k)\prod_{i=0}^{k-1}(x-x_i) \quad (٢٣-٧)$$

يمكن وضع معادلة نيوتن للاستيفاء فى صورة جدول يسهل من عمليات الحساب اليدوى لها كما فى جدول ٧-١ الذى تم إنشاؤه لحساب معادلة نيوتن لاستيفاء ٤ نقاط هى (x_0,y_0) و (x_1,y_1) و (x_2,y_2) و (x_3,y_3). العمود الأول (اليسار) فى هذا الجدول هو المركبة x للنقاط الأربعة. العمود الثانى هو المركبة y. بعد ذلك تم حساب قسمة الفروق من الدرجة الأولى فى العمود الثالث من محتويات العمود الثانى. ثم تم حساب قسمة الفروق من الدرجة الثانية فى العمود الرابع باستخدام قسمة الفروق من الدرجة الأولى التى فى العمود الثالث. وفى النهاية تم حساب قسمة الفروق من الدرجة الثالثة فى العمود الخامس باستخدام قسمة الفروق من الدرجة الثانية التى فى العمود الرابع. لاحظ أن الجدول الناتج عبارة عن مصفوفة قطرية.

جدول ٧-١ قسمة الفروق من الدرجات الأولى والثانية والثالثة لأربع نقاط فى صورة جدولية

x	Y=f(x)	1st d. diff.	2nd d. diff.	3rd d. diff.
x_0	$y_0=f(x_0)$			
x_1	$y_1=f(x_1)$	$f(x_0.x_1) = \frac{y_1-y_0}{x_1-x_0}$		
x_2	$y_2=f(x_2)$	$f(x_1.x_2) = \frac{y_2-y_1}{x_2-x_1}$	$f(x_0.x_1.x_2) = \frac{f(x_1.x_2)-f(x_0.x_1)}{(x_2-x_0)}$	
x_3	$y_3=f(x_3)$	$f(x_2.x_3) = \frac{y_3-y_2}{x_3-x_2}$	$f(x_1.x_2.x_3) = \frac{f(x_2.x_3)-f(x_1.x_2)}{(x_3-x_1)}$	$f(x_0.x_1.x_2.x_3) = \frac{f(x_1.x_2.x_3)-f(x_0.x_1.x_2)}{(x_3-x_0)}$

بعد الانتهاء من هذا الجدول يمكن كتابة معادلة نيوتن للاستيفاء لهذه النقاط الأربعة باستخدام عناصر القطر الرئيسى من هذا الجدول الناشىء.

$$P(x) = f(x_0) + f(x_0.x_1)(x - x_0) + f(x_0.x_1.x_2)(x - x_0)(x - x_1) + f(x_0.x_1.x_2.x_3)(x - x_0)(x - x_1)(x - x_2) \quad (٧-٢٤)$$

سنوضح ذلك بالمثال التالى:

مثال ٧-٥

استخدم طريقة نيوتن لإيجاد كثيرة الحدود التى تستوفى النقاط الأربعة التالية: (3 ,2-) و (0 ,1) و (1- ,0) و (2 ,1-). جدول ٧-٢ يبين قسمة الفروق من الدرجات الأولى والثانية والثالثة لهذه النقاط الأربعة:

جدول ٧-٢ قسمة الفروق للنقاط الأربعة فى مثال ٧-٥

x	Y=f(x)	1st d. diff.	2nd d. diff.	3rd d. diff.
-2	3			
1	0	$\frac{0-3}{1+2} = -1$		
0	-1	$\frac{-1-0}{0-1} = 1$	$\frac{1+1}{0+2} = 1$	
-1	2	$\frac{2+1}{-1-0} = -3$	$\frac{-3-1}{-1-1} = 2$	$\frac{2-1}{-1+2} = 1$

وعلى ذلك يمكن استخدام عناصر القطر الرئيسى فى كتابة كثيرة الحدود من المعادلة (٧- ٢٤) كما يلى:

$$P(x) = 3 - 1(x - x_0) + 1(x - x_0)(x - x_1) + 1(x - x_0)(x - x_1)(x - x_2)$$

$$P(x) = 3 - (x + 2) + (x + 2)(x - 1) + (x + 2)(x - 1)(x) \quad (٧-٢٥)$$

مثال ٧-٦

استخدم طريقة نيوتن لإيجاد كثيرة الحدود التى تستوفى النقاط الخمسة التى فى الجدول التالى:

x_i	0	1	2	3	4
y_i	0	1	8	27	64

جدول ٧-٣ يبين قسمة الفروق من الدرجات الأولى والثانية والثالثة والرابعة لهذه النقاط الخمسة:

وعلى ذلك يمكن استخدام عناصر القطر الرئيسى فى كتابة كثيرة الحدود من المعادلة (٧- ٢٤) كما يلى:

$$P(x) = 0 + 1(x - x_0) + 3(x - x_0)(x - x_1) + 1(x - x_0)(x - x_1)(x - x_2) + 0(x - x_0)(x - x_1)(x - x_2)(x - x_3$$

$$P(x) = x + 3x(x - 1) + x(x - 1)(x - 2) \quad (٧-٢٦)$$

جدول ٧-٣ قسمة الفروق للنقاط الأربعة فى مثال ٧-٥

x	Y=f(x)	1st d. diff.	2nd d. diff.	3rd d. diff.	4th d. diff.
0	0				
1	1	$\frac{1-0}{1-0}=1$			
2	8	$\frac{8-1}{2-1}=7$	$\frac{7-1}{2-0}=3$		
3	27	$\frac{27-8}{3-2}=19$	$\frac{19-7}{3-1}=6$	$\frac{6-3}{3-0}=1$	
4	64	$\frac{64-27}{4-3}=37$	$\frac{37-19}{4-2}=9$	$\frac{9-6}{4-1}=1$	$\frac{1-1}{4-0}=0$

٧-٦ تمثيل كثيرة حدود نيوتن بطريقة هورنر

الصورة النهائية لكثيرة حدود نيوتن من الدرجة n موجودة فى المعادلة (٧- ٢٢) وسنعيد كتابتها هنا كما يلى:

$$P(x) = f(x_0) + f(x_0.x_1)(x-x_0) + f(x_0.x_1.x_2)(x-x_0)(x-x_1) + f(x_0.x_1.x_2.x_3)(x-x_0)(x-x_1)(x-x_2) + \cdots + f(x_0.x_1 \dots .x_n)(x-x_0)(x-x_1)\dots(x-x_{n-1}) \qquad (٢٧-٧)$$

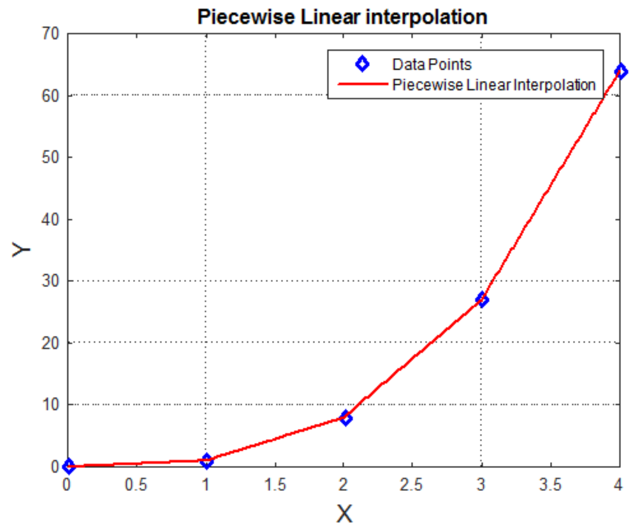
هذه المعادلة يمكن كتابتها بطريقة هورنر كما يلى:

$$P(x) = f(x_0) + (x-x_0)\Big(f(x_0.x_1) + (x-x_1)\Big(f(x_0.x_1.x_2) + (x-x_2)\big(f(x_0.x_1.x_2.x_3) + \cdots\ (x-x_{n-2})\big(f(x_0.x_1 \dots .x_{n-1}) + (x-x_{n-1})(f(x_0.x_1 \dots .x_n))\big)\big)\Big)\dots\Big) \qquad (٢٨-٧)$$

وهذا بالطبع سيكون له الأثر الكبير فى حالة حساب قيمة كثيرة الحدود عند أى قيمة للمتغير x حيث سيتم توفير الكثير من الوقت.

البرنامج التالى عبارة عن برنامج مقترح وبسيط لحساب الفروق المقسومة لأى عدد من النقاط:

```
% Calculation of divided diff for Newton method
% div_diff function computes all divided differences for
%  the data stored in x and y = f(x)
x = input([' Enter The Array of Coefficients x= ']); %e.g [1;2;]
y = input([' Enter The Array of Coefficients y= ']); %e.g [1;2;]
n = length(x);
m = length(y);
if m~=n; error('x and y must be same size');
else
       F = zeros(n, n);
       for i = 1:n  % define 0th divide difference
          F(i,1) = y(i);
       end
       for k = 1:n-1   % Get kth divided difference
          for i = 1:n-k
             F(i,k+1) = (F(i+1,k)-F(i,k))/(x(i+k)-x(i));
          end
       end
```



شكل ٧-١٠ الاستيفاء الخطى كثير المقاطع

```
        end
        disp('Divided differences will be the first row of F');
        F(1,1:n) %div. diff components.
```

بتنفيذ هذا البرنامج على نقاط المثالين ٧-٥ حصلنا على ما يلى:

\>> NewtonInterp

Enter The Array of Coefficients x= [-2;1;0;-1]

Enter The Array of Coefficients y= [3;0;-1;2]

Divided differences will be the first row of F

ans =

3 -1 1 1

وبتنفيذه على النقاط الخمسة التى فى المثال ٧-٦ نحصل على:

\>> NewtonInterp

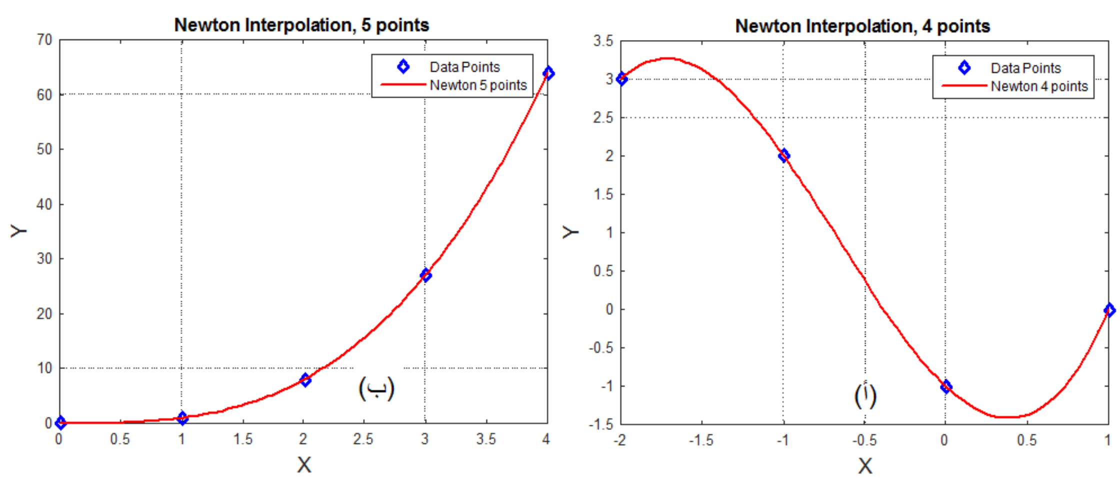
Enter The Array of Coefficients x= [0;1;2;3;4]

Enter The Array of Coefficients y= [0;1;8;27;64]

Divided differences will be the first row of F

ans =

0 1 3 1 0



شكل ٧-٩ استيفاء نيوتن (أ) للنقاط الأربعة التى فى مثال ٧-٥ و (ب) النقاط الخمسة التى فى المثال ٧-٦

بالتويض بهذه الفروق المقسومة فى كثيرة حدود نيوتن لهذه الأربع نقاط التى فى المعادلة (٧- ٢٥) نحصل على استيفاء نيوتن لهذه النقاط الأربعة التى فى مثال ٧-٥ كما فى شكل ٧-٩أ، واستيفاء نيوتن للنقاط الخمسة التى فى مثال ٧-٦ كما فى شكل ٧-٩ب.

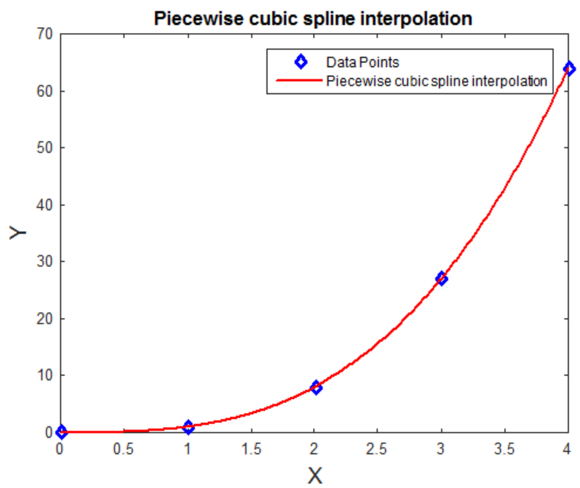
٧-٧ الاستيفاء بكثيرات الحدود متعددة المقاطع piecewise

لقد تم استخدام كثيرات الحدود فى الأجزاء السابقة من أجل استيفاء نقط البيانات أو تقريب الدوال المركبة. ولقد رأينا أنه كلما زادت درجة كثيرة الحدود (بسبب زيادة عدد النقاط أو الحصول على دقة أعلى عند تقريب أى دالة) فإن كثيرة الحدود تميل إلى التذبذب بين نقاط البيانات ويزداد هذا التذبذب عن الأطراف بدرجة كبيرة. للتغلب على هذه المشكلة فإنه يتم اللجوء إلى تقسيم مجموعة النقاط الكبيرة المراد استيفاؤها أو المنحنى المراد تقريبه إلى مقاطع أو الأقسام يتم استيفاء كل منها على حده ثم التوصيل فيما بينها. ولذلك فإن هذه الطريقة تسمى بالطريقة متعددة المقاطع، ولقد رأينا أبسط هذه الأنواع فى الجزء ٧-١ الخاص بالتقريب الخطى حيث رأينا أن دالة ماتلاب جاهزة البناء interp1(x,y,'linear') تقوم بهذه المهمة تماما حيث تقوم بالتوصيل فيما بين كل نقطتين متتاليتين بخط مستقيم، ولقد قمنا بتطبيق ذلك على الخمسة نقاط التى فى المثال ٧-٦ كما هو موضح فى شكل ٧-١٠. هذه الطريقة مع أنها لا يكون بها تذبذب على الإطلاق لدالة الاستيفاء سواء فيما بين النقاط أو على أطراف الدالة إلا أن بها عيبا أساسيا وهو أن كثيرة حدود الاستيفاء تكون متقطعة أو منكسرة أو غير انسيابية عند نقاط البيانات أو عند الانتقال من مقطع إلى مقطع، وهذا يعنى أن كثيرة الحدود لن تكون قابلة للتفاضل عند هذه النقاط. فى بعض الأحيان تستخدم هذه الطريقة إذا كانت نقاط البيانات قريبة جدا من بعضها.

لذلك كان الحل هو استيفاء كل مقطع من هذه المقاطع بكثيرة حدود من الدرجة الثالثة كما سنرى. بفرض n+1 من النقاط المطلوب استيفاؤها وهى (x_0,y_0) و (x_1,y_1) و (x_2,y_2) و ... (x_n,y_n)، وهذه النقاط يطلق عليها أحيانا العقد. سنفترض أن هذا المدى من النقاط مقسم إلى عدد n من المقاطع كل مقطع يصل بين نقطتين متتاليتين وسيتم استيفاء كل مقطع من هذه المقاطع بمنحنى أو كثيرة حدود من الدرجة الثالثة كما يلى:

$$p_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i \qquad (٧-٢٩)$$

حيث i تتغير من صفر حتى n−1 و $p_i(x)$ تمثل المنحنى من الدرجة الثالثة داخل كل مقطع من هذه المقاطع، وأن منحنى الاستيفاء العام $p(x)$ يجب أن يمر بجميع هذه النقاط أوالعقد (n+1) ويحقق أيضا هذا المنحنى داخل كل مقطع وبالتالى فإن $p(x)=p_i(x)$ لجميع قيم x داخل المقطع i.

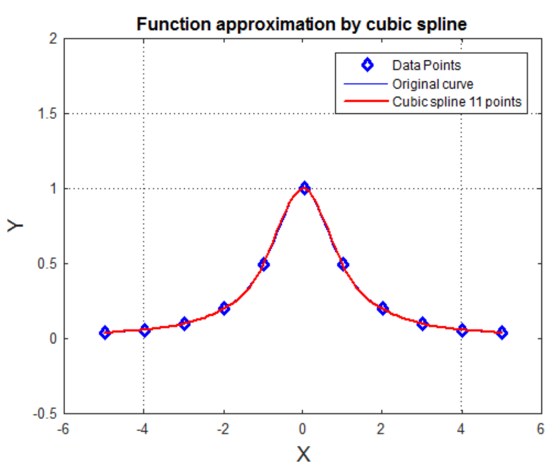


شكل ٧−١١ استيفاء نقاط المثال ٧−٦ باستخدام الاستيفاء المقطعى من الدرجة الثالثة

لتحقيق هذه الانسيابية بين مقاطع هذه المنحنيات فلابد من وضع هذه الشروط التالية على المعادلة (٧− ٢٩):

١− لكى تتحقق الانسيابية لكثيرة الحدود العامة $p(x)$ عند جميع العقد فإن قيمة كثيرة الحدود العامة عند أى عقدة يجب أن تحقق الشرط $p(x_i)=y_i$ لجميع قيم i من صفر حتى n−1.

٢− عند أى عقدة i يجب أن تكون قيمة كثيرة الحدود فى المقطع السابق للنقطة i تساوى قيمة كثيرة الحدود فى المقطع التالى للعقدة i، بمعنى $p_{i-1}(x_i)=p_i(x_i)$ لجميع قيم i من 1 حتى n−1.



شكل ٧−١٠ تقريب الدالة 1/(1+x2) باستخدام الاستيفاء الثلاثى متعدد المقاطع حيث نلاحظ عدم وجود أى تذبذبات

٣− يجب أن يكون التفاضل الأول مستمرا أيضا عند كل عقدة، وبالتالى فإن $p'_{i-1}(x_i) = p'_i(x_i)$ لجميع قيم i من 1 حتى n−1.

يجب أن يكون التفاضل الثانى مستمرا أيضا عند كل عقدة، وبالتالى فإن $p''_{i-1}(x_i) = p''_i(x_i)$ لجميع قيم i من 1 حتى n−1.

بالتعويض بهذه الشروط الأربعة فى المعادلة (٧− ٢٩) يمكن الحصول على مجموعة من المعادلات فى المجاهيل ai و bi و ci و di التى يمكن حلها والحصول على قيم هذه المجاهيل. وحيث أن هذه العملية تحتاج للكثير من الاختصارات الرياضية التى لا داعى للدخول فى تفاصيلها، فإننا سنكتفى بالفكرة التى ذكرناها هنا.

يحتوى ماتلاب على الدالة pp=spline(x,y) التى تقوم باستيفاء مجموعة النقاط (x, y) مستخدمة متعدد المقاطع من الدرجة الثالثة. كمثال على ذلك فقد قمنا باستخدام الاستيفاء متعدد المقاطع من الدرجة الثالثة لاستيفاء النقاط الخمسة التى فى المثال ٧-٦ كما فى شكل ٧-١١. والبرنامج المستخدم فى ذلك هو:

```
1- %Piecewise cubic spline interpolation
2- x = [0 1 2 3 4];
3- y = [0 1 8 27 64];
4- plot(x,y,'bd','Linewidth',2); hold on;
5- x_new = 0:.01:4;
6- pp = spline(x,y);
7- y_new=ppval(pp,x_new);
8- plot(x_new,y_new,'r', 'linewidth',1.5);
9- hold on; grid;
10- title('Piecewise cubic spline interpolation','fontsize',12)
11- ylabel('Y','fontsize',14)
12- xlabel('X','fontsize',14)
13- legend('Data Points','Piecewise cubic spline interpolation')
```

الأمر رقم ٦ يحسب استيفاء النقاط الخمسة ويضع معاملات هذا الاستيفاء فى المتغير pp، فى الأمر ٧ تم استخدام الدالة ppval(pp,x_new) للتعويض بالمركبات x_new فى كثيرة الحدود التى معاملاتها هى pp. يمكن استخدام الأمرين ٦ و ٧ فى أمر واحد كالتالى: y_new=spline(x,y,x_new) حيث يتم حساب pp والتعويض مباشرة بقيم x_new للحصول على y_new.

لكى نرى تأثير الاستيفاء الثلاثى متعدد المقاطع على التذبذب الذى كان يحدث مع استخدام الطرق السابقة أحادية المقطع مثل لاجرانج أو نيوتن أو فاندرموند تعال نستخدم الاستيفاء الثلاثى متعدد المقاطع على تقريب الدالة f(x)=1/(1+x2) التى قربناها باستخدام طريقة لاجرانج كما فى شكل ٧-٧ب باستخدام ١١ نقطة تقريب ورأينا فى هذا الشكل مدى تذبذب الدالة المقربة عند أطراف مدى التقريب. شكل ٧-١٢ يبين نفس هذه الدالة مقربة باستخدام الاستيفاء الثلاثى متعدد المقاطع باستخدام ١١ نقطة أيضا حيث نرى أن منحنى التقريب (الأحمر) ينطبق تماما على المنحنى الأصلى للدالة (الأزرق) لذلك فإن المنحنى الأزرق غير ظاهر مع عدم وجود أى تذبذبات على الأطراف على الإطلاق مما يوضح ميزة استخدام هذه الطريقة فى تقريب الدوال.

البرنامج المستخدم فى الحصول على الشكل ٧-١٢ هو:

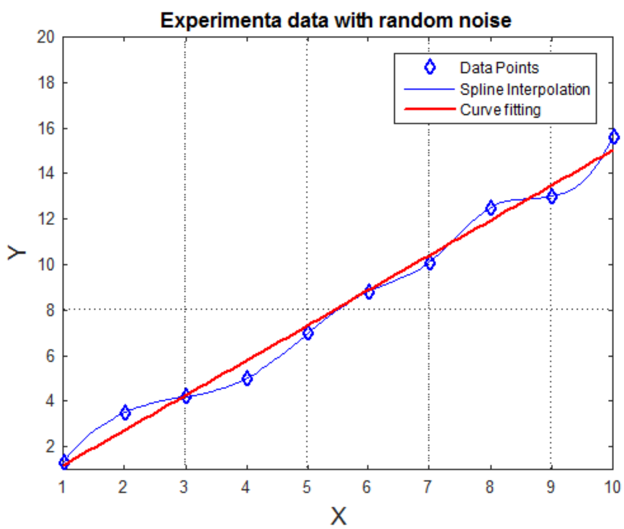
```
%Piecewise cubic spline interpolation
pointx = linspace(-5,5,11);
pointy = 1./(1+pointx.^2);
x_new = linspace(-5,5);
y_new = 1./(1+x_new.^2);
plot(pointx,pointy,'bd','Linewidth',2); hold on;
plot(x_new,y_new,'b');hold on;
y_news=spline(pointx,pointy,x_new);
plot(x_new,y_news,'r', 'linewidth',1.5);
```

```
hold on; grid;
title('Function approximation by cubic spline','fontsize',12)
ylabel('Y','fontsize',14)
xlabel('X','fontsize',14)
legend('Data Points','Original curve','Cubic spline 11 points')
axis([-6 6 -0.5 2]);
```

يمكن استخدام الأمر y_new=interp1(x,y,x_new,'spline') للحصول على نفس تأثير الأمر spline الذى فى البرنامج السابق. لاحظ أن الدالة interp1( ) قد استخدمناها مع الاستيفاء الخطى حيث كتبنا المعامل 'linear' بدلا من 'spline'، وفى الحقيقة فإن هذه الدالة تستخدم مع أكثر من نوع من أنواع الاستيفاء. يمكنك كتابة help interp1 فى نافذة التفاعل فى ماتلاب لترى الكثير من المعلومات عن هذه الدالة.

٧-٨ تقريب المنحنيات Curve fitting

هناك نوعان من المشاكل نحتاج فيهما لتقريب المنحنيات. المشكلة الأولى عندما يكون لدينا دالة معقدة ونريد تمثيلها أو تقريبها بدالة أو منحنى أكثر بساطة وهذا ما رأيناه كأحد تطبيقات الاستيفاء التى شرحناها فى لأجزاء السابقة. المشكلة الأخرى والتى تعنينا هنا هى عندما يكون لدينا بيانات ناتجة من تجربة معينة وهذه البيانات بها بعض الضوضاء العشوائية المضافة ونحن نريد تقريب هذه البيانات بمنحنى أو دالة لا تمر بجميع هذه النقاط المحتوية على الضوضاء ولكنها تمر فى مسار متوسط بين هذه النقاط بحيث يكون الخطأ أو الفرق بين هذه النقاط والمنحنى التقريبى أقل ما يمكن وأن تكون درجة هذا المنحنى أقل ما يمكن. لكى نوضح ذلك افترض القراءات الناتجة من أحد التجارب والموضحة فى جدول ٧-٤، والتى تم رسمها فى شكل ٧-١٣.



شكل ٧-١٣ استيفاء وتقريب بيانات الجدول ٧-٤

جدول ٧-٤ بيانات أحد التجارب المعملية

X	1	2	3	4	5	6	7	8	9	10
Y	1.3	3.5	4.2	5	7	8.8	10.1	12.5	13	15.6

كما نرى من شكل ٧-١٤ فإن اللجوء إلى استيفاء نقاط التجربة حتى بأفضل طرق الاستيفاء التى تعطى أقل تذبذب فإن المنحنى الناتج من هذا الاستيفاء لا يكون مرضيا من حيث الشكل فى الكثير من التطبيقات كما أنه لا يعكس

سلوك تغير هذه التجربة مع تغير المركبة x. الشكل العام لهذه النقاط يبين أن هناك علاقة خطية بين متغيرى هذه التجربة ولذلك فإننا نفضل رسم خط مستقيم يتوسط هذه النقاط بحيث يكون الخطأ بين هذا الخط المستقيم ونقاط التجربة أقل ما يمكن. كما أننا نريد من هذا المنحنى التقريبي أن يكون بأقل درجة كما ذكرنا.

على وجه العموم بفرض أن لدينا نقاط البيانات (x_i,y_i) حيث i تتغير من 1 إلى n، أى أن لدينا عدد n من النقاط. المطلوب هو البحث عن دالة $a_jf(x)$ حيث j تغير من 1 إلى m وعلى أن تكون m<n أى أن الدالة f(x) لها عدد من المعاملات أقل من عدد نقاط البيانات المطلوب تقريبها، وهذه الدالة يجب أن تعطى أفضل تقريب. لكى تعطى هذه الدالة أفضل تقريب فإننا نضع عليها شرط أن يكون مجموع الجذر مربعات الخطأ بينها وبين جميع نقاط البيانات أقل ما يمكن. يمكن كتابة هذا الخطأ كما يلى:

$$E(a_1.a_2 \dots .a_m) = \sum_{i=1}^{n}(y_i - f(x_i))^2 \qquad (٧-٣٠)$$

هذا الخطأ الذى فى المعادلة (٧- ٣٠) نريده أقل ما يمكن، ولذلك فإن هذه الطريقة تسمى طريقة أقل المربعات least square method. أى أننا نريد البحث عن معاملات الدالة a_j التى تجعل مجموع المربعات التى فى المعادلة (٧- ٣٠) أقل ما يمكن. لكى نحصل على ذلك فإننا نفاضل المعادلة (٧- ٣٠) بالنسبة لكل هذه المعاملات aj ونساوى نتيجة هذا التفاضل بالصفر كما يلى:

$$\frac{\partial E}{\partial a_j} = 0 \qquad (٧-٣١)$$

، حيث j تتغير من 1 إلى m.

المعادلة (٧- ٣١) هى فى الحقيقة عدد m من المعادلات التى فى الغالب ما تكون غير خطية فى المعاملات a_j ويكون الحصول على حل لها صعبا. الدالة f(x) تسمى دالة الأساس والصورة المبسطة لمعادلة التقريب ستكون هى الصورة الخطية التالية:

$$f(x) = a_1f_1(x) + a_2f_2(x) + \cdots a_mf_m(x) \qquad (٧-٣٢)$$

وهناك الكثير من دوال الأساس التى يمكن استخدامها ومنها مثلا، $f(x)=x^k$ و $f(x)=\cos(kx)$ و $f(x)=e^{-kx}$ و $f(x)=a+bx$.

التقريب باستخدام خط مستقيم

سنفترض أن الدالة f(x) عبارة عن خط مستقيم على الصورة f(x)=a+bx. فى هذه الحالة سيكون الخطأ المراد تقليله إلى أقل درجة كما فى المعادلة (٧- ٣٠) هو:

$$E(a.b) = \sum_{i=1}^{n}(y_i - a - bx_i)^2 \qquad (٧-٣٣)$$

بتفاضل هذا الخطأ بالنسبة لكل من a و b ومساوتهما بالصفر نحصل فى النهاية على قيمة كل من a و b كما يلى:

$$a = \frac{(\sum_{i=1}^{n} x_i^2)(\sum_{i=1}^{n} y_i^2) - (\sum_{i=1}^{n} x_i y_i)(\sum_{i=1}^{n} x_i)}{n(\sum_{i=1}^{n} x_i^2) - (\sum_{i=1}^{n} x_i)^2} \quad (٧-٣٤)$$

$$b = \frac{n(\sum_{i=1}^{n} x_i y_i) - (\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{n(\sum_{i=1}^{n} x_i^2) - (\sum_{i=1}^{n} x_i)^2} \quad (٧-٣٥)$$

بالتعويض من الجدول ٧-٤ فى المعادلتين (٧- ٣٤) و ٧- ٣٥) نجد أن a=-0.360 و b=1.538 وبالتالى فإن المعادلة الخطية ستكون f(x)=1.538x-0.360.

التقريب باستخدام كثيرات الحدود

باستخدام دالة الأساس فى المعادلة (٧- ٣٢) كقوى للمتغير x نحصل على معادلة التقريب فى صورة كثيرة حدود كما يلى:

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

حيث m<n. فى هذه الحالة ستكون معادلة مجموع مربع الخطأ عند كل نقاط البيانات n كما يلى:

$$E(a_0.a_1 \dots .a_m) = \sum_{i=1}^{n} (y_i - P(x_i))^2$$

لتقليل هذا الخطأ لأقل ما يمكن نقوم بتفاضل هذا الخطأ بالنسبة لكل المعاملات ونساوى بالصفر $\partial E/\partial a_j=0$ لكل قيم j من 1 حتى m فنحصل على m+1 من المعادلات فى المجاهيل a_0 حتى a_m كما يلى:

$$a_0 \sum_{i=1}^{n} x_i^0 + a_1 \sum_{i=1}^{n} x_i^1 + a_2 \sum_{i=1}^{n} x_i^2 + \cdots + a_m \sum_{i=1}^{n} x_i^m = \sum_{i=1}^{n} y_i x_i^0 \quad (٧-٣٦)$$

$$a_0 \sum_{i=1}^{n} x_i^0 + a_1 \sum_{i=1}^{n} x_i^1 + a_2 \sum_{i=1}^{n} x_i^2 + \cdots + a_m \sum_{i=1}^{n} x_i^m = \sum_{i=1}^{n} y_i x_i^0$$

$$a_0 \sum_{i=1}^{n} x_i^1 + a_1 \sum_{i=1}^{n} x_i^2 + a_2 \sum_{i=1}^{n} x_i^3 + \cdots + a_m \sum_{i=1}^{n} x_i^{m+1} = \sum_{i=1}^{n} y_i x_i^1$$

وهكذا

$$a_0 \sum_{i=1}^{n} x_i^m + a_1 \sum_{i=1}^{n} x_i^{m+1} + a_2 \sum_{i=1}^{n} x_i^{m+2} + \cdots + a_m \sum_{i=1}^{n} x_i^{2m} = \sum_{i=1}^{n} y_i x_i^m$$

هذه المعادلات سيكون لها حل طالما أن xi متفردة، أى ليس هناك نقطتان بنفس المركبة x لأن ذلك يعنى خط رأسى أو انكسار كما رأينا فى حل المعادلات الخطية فى الفصلين ٣ و ٤.

مثال ٧-٧

أوجد أفضل منحنى تقريب من الدرجة الثانية لنقاط البيانات التالية:

x_i	0	0.25	0.50	0.75	1.00
y_i	1.0000	1.2840	1.6487	2.1170	2.7183

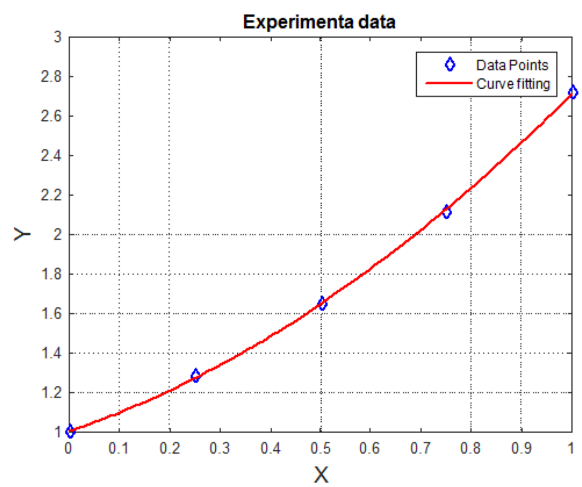
بالتعويض فى مجموعة المعادلات (٧- ٣٦) عن n=5 و m=2 نحصل على المعادلات التالية:

$$5a_0 + 2.5a_1 + 1.875a_2=8.7680$$
$$2.5a_0 + 1.875a_1 + 1.5625a_2=5.4514$$
$$1.875a_0 + 1.5625a_1 + 1.3828a_2=4.4015$$

بحل هذه المعادلات الثلاثة فى الثلاث مجاهيل نحصل على $a_0=1.0051$ و $a_1=0.86468$ و $a_2=0.84316$ وبذلك تكون معادلة المنحنى من الدرجة الثانية التى تعطى أفضل تقريب لهذه النقاط هى:

$y=1.0051+0.86468x+0.84316x^2$.

بتطبيق برنامج جاوس سايدل فى الفصل ٣ لحل المعادلات السابقة حصلنا على نفس القيم كما يلى:



شكل ٧-١٤ منحنى التقريب للنقاط الخمسة التى فى المثال ٧-٧

```
>> GaussSeidel
Enter The Array of Coefficients a= [5 2.5 1.875;2.5 1.875 1.5625;1.875 1.5625 1.3828]
Enter The Arrays of Constants b = [8.7680;5.4514;4.4015]
Enter The Array of first trial x0 = [0;0;0]
Enter The Max number of iterations = 500
Enter The Error tolerance = 0.000001
   1.0051    0.8647    0.8431
```

البرنامج التالى سيرسم هذه النقاط ويرسم عليها المنحنى الناتج من الدرجة الثانية:

```
% curve fitting using matlab commands
x=[0 0.25 0.5 0.75 1];
y=[1.000 1.2840 1.6487 2.1170 2.7183];
xx=linspace(0,1);
plot(x,y,'bd', 'linewidth', 1.5);
hold on; grid;
c=polyfit(x,y,2)
yvals=polyval(c,xx);
plot(xx,yvals,'r','linewidth',1.5);
title('Experimenta data','fontsize',12)
ylabel('Y','fontsize',14)
xlabel('X','fontsize',14)
legend('Data Points','Curve fitting')
axis([0 1 1 3]);
```

فى هذا البرنامج الأمر رقم 7، c=polyfit(x,y,2) يقوم بعمل تقريب للنقاط x و y بمنحنى من الدرجة الثانية (الرقم 2 فى الأمر يعنى ذلك، ويمكن وضع أى درجة تكون أقل من عدد النقاط) ويضع معاملات هذا المنحنى فى المتغير c. نتيجة تنفيذ هذا البرنامج كانت كالتالى، والمنحنى الناتج موضح فى شكل ٧-١٤.

```
c =
   0.8437    0.8642    1.0051
```

وهى نفس القيم التى حصلنا عليها باستخدام طريقة جاوس سايدل وعن طريق الحل اليدوى كما سبق.

٧-٩ تمارين

١- استخدم طريقة لاجرانج من الدرجة الأولى ومن الدرجة الثانية مرة أخرى لاستيفاء الدوال التالية: (١) $f(x)=\cos(x)$ و (٢) $f(x)=\sqrt{1+x}$ و (٣) $f(x)=\ln(x+1)$ و (٤) $f(x)=\tan(x)$ عند القيم $x_0=0$ و $x_1=0.6$ و $x_2=0.9$. أوجد القيمة التقريبية $f(0.55)$ و $f(0.65)$ فى كل حالة.

٢- أعد تمرين ١ مستخدما طريقة نيوتن.

٣- استخدم طريقة لاجرانج من الدرجة الثالثة لتقريب الدالة $f(x)=\log_{10}(\tan(x))$ عند قيم $x=1.00$ و $x=1.05$ و $x=1.10$ و $x=1.15$ ثم أوجد قيمة $f(1.09)$ من الدالة المقربة وأوجد مقدار الخطأ بين هذه القيمة وقيمتها الحقيقية.

٤- أعد مستخدما طريقة نيوتن.

٥- إفترض أن الدالة $P_3(x)$ هى دالة لاجرانج لاستيفاء نقاط البيانات (0,0) و (0.5,y) و (1,3) و (2,2). أوجد قيمة y إذا كان معامل x^3 فى الدالة $P_3(x)$ يساوى 6.

٦- الجدول التالى يبين تعداد السكان فى أحد البلدان من عام ١٩٤٠ حتى عام ١٩٩٠، أوجد كثيرة حدود لاجرانج من الدرجة الخامسة لاستيفاء هذه البيانات. أوجد تقدير لتعداد السكان فى الأعوام ١٩٣٠ و ١٩٦٥ و ١٩١٠. ما هى دقة التقدير فى عام ١٩٣٠ إذا كان التعداد الحقيقى هو 123203000.

جدول تمرين ٦

السنة	1940	1950	1960	1970	1980	1990
التعداد	132165	151326	179323	203302	226542	249633

٧- أعد تمرين ٦ مستخدما طريقة نيوتن للاستيفاء.

٨- أوجد دالة استيفاء نيوتن من الدرجة الرابعة لنقاط البيانات التى فى جدول تمرين ٨:

جدول تمرين ٨

x	0.0	0.1	0.3	0.6	1.0
f(x)	-6.00000	-5.89483	-5.65014	-5.17788	-4.28172

٩- فى التمرين ٨ إذا إضيفت النقطة f(1.1)=-3.99583 إلى الجدول، أوجد كثيرة حدود نيوتن من الدرجة الخامسة.

١٠- استخدم طريقة نيوتن لاستيفاء نقاط جدول تمرين ٩ واحسب القيمة التقريبية f(0.3):

جدول تمرين ٩

x	0.0	0.2	0.4	0.6
f(x)	15.0	21.0	30.0	51.0

١١- افترض أنه تم اكتشاف أن f(0.4) كانت زائدة بمقدار 10 وأن f(0.6) كانت ناقصة بمقدار 5، فماذا سيحدث للقيمة المقدرة f(0.3) على ضوء هذا التغيير؟

١٢- نقاط البيانات التى فى جدول تمرين ١٢ تقع على منحنى الدالة $f(x)=4.8\cos(\pi x/20)$. استخدم طريقة لاجرانج لاستيفاء هذه النقاط. أوجد قيمة دالة لاجرانج للاستيفاء عند $x=0, 0.5, 1, \ldots, 8$ وقارن هذه النتائج مع القيم الصحيحة المحسوبة من التعويض المباشر فى المعادلة f(x).

جدول تمرين ١٢

x	0.15	2.30	3.15	4.85	6.25	7.95
y	4. 79867	4. 49013	4. 2243	3. 47313	2. 66674	1. 51909

١٣- أعد التمرين السابق مستخدما طريقة نيوتن.

١٤- أكتب ملف دالة فى ماتلاب يحسب دوال الأساس للاستيفاء عن طريق لاجرانج حيث يتم النداء على هذه الدالة من البرنامج الأساسى بعد إدخال النقاط المراد استيفاؤها وعددها. استعن بخريطة التدفق الموضحة فى شكل ٧-٨.

١٥- أكتب ملف دالة فى ماتلاب يحسب استيفاء نيوتن لعدد n من النقاط حيث يتم النداء على هذه الدالة من البرنامج الأساسى بعد إدخال النقاط المراد استيفاؤها وعددها، ثم يقوم البرنامج برسم دالة الاستيفاء، وإعطاء الفروق المقسومة.

١٦- استخدم طريقة لاجرانج لاستيفاء نقاط الجدول التالى:

جدول تمرين ١٦

x	2	1	4	-1	3	-4
y	-1	2	59	4	24	-53

١٧- أعد التمرين السابق مستخدما طريقة نيوتن.

١٨- إرسم المنحنيات التقريبية من الدرجة الأولى والثانية والثالثة لنقاط البيانات التى فى الجدول التالى:

جدول تمرين ١٨

x	1.0	1.1	1.3	1.5	1.9	2.1
y	1.84	1.96	2.21	2.45	2.94	3.18

إحسب الخطأ فى كل حالة.

١٩- أعد التمرين ١٨ لجدول البيانات التالى:

جدول تمرين ١٩

x	0	0.15	0.31	0.5	0.6	0.75
y	1	1.004	1.031	1.117	1.223	1.422

٢٠- أعد التمرين السابق لجدول البيانات التالى:

جدول تمرين ٢٠

x	4	4.2	4.5	4.7	5.1	5.5	5.9	6.3	6.8	7.1
y	102.56	113.18	130.11	142.05	167.53	195.14	224.87	256.73	299.50	326.72

الفصل ٨

التفاضل العددى والتكامل العددى

Numerical Differentiation and Integration

الفصل ٨

التفاضل العددى والتكامل العددى

Numerical Differentiation and Integration

٨–١ مفاهيم أساسية عن التفاضل العددى

المشكلة التى سنتناولها هنا هى أن لدينا دالة $y=f(x)$ ونريد الحصول على تفاضل هذه الدالة عند النقطة $x=x_i$. هذه الدالة $f(x)$ إما أنها تكون متاحة فى صورة يمكن تحويلها إلى مجموعة من النقاط (x_i,y_i) حيث i تتغير من 1 إلى n، أو أنها تكون معطاة أصلا فى هذه الصورة من النقاط. إن ذلك يجيب على سؤال يتبادر إلى الذهن بأن التفاضل للدوال فى صورتها التحليلية يكون فى العادة أسهل من التكامل وربما لا تكون هناك حاجة للبحث عن التفاضل العددى مثلما هو الحال بالنسبة للتكامل العددى الذى سندرسه فى الجزء الثانى من هذا الفصل. وهذا صحيح فعلا ولكن وجود أى علاقة عملية بين متغيرين لا تكون دائما فى صورة تحليلية، ولكنها تكون فى الغالب فى صورة نقاط منفصلة، وهنا تكون الحاجة إلى التفاضل العددى. هنا أيضا تظهر فكرة أخرى حيث من الممكن أن يقول البعض بأنه يمكننا أن نقوم باستيفاء هذه النقاط بأحد طرق الاستيفاء التى تمت دراستها ثم نقوم بحساب التفاضل لهذه الدالة المستوفاة، وهذه هى فعلا ستكون أحد الطرق المقترحة لحساب التفاضل العددى. يجب أن نتوقع أن حساب التفاضل العددى سيخضع لبعض الخطأ مثله فى ذلك مثل جميع الطرق العددية وأنه من المستحيل الحصول على نفس الدقة التى يمكن الحصول عليها من تفاضل الصورة التحليلية للدالة.

نذكر هنا أن ذلك يختلف تماما عن إمكانية التفاضل والتكامل على التعبيرات الرمزية التى يجريها ماتلاب من خلال صندوق أدوات خاص بذلك وهو صندوق أدوات الحسابات الرمزية symbolic math toolbox، والذى عن طريقه يمكن حساب التفاضل والتكامل بنفس الصور والقوانين التحليلية التى نعرفها ولقد نوهنا عن هذا الصندوق فى الفصل الخاص بمراجعة ماتلاب.

٨–٢ التفاضل الفرقى

التقريب المبسط للتفاضل الأول يمكن كتابته كما يلى:

$$f'(x) \approx \frac{f(x+h)-f(x)}{h} \qquad (١-٨)$$

حيث من المفترض أن تكون h صغيرة ولكنها لا تساوى الصفر. ويمكن حساب الخطأ فى هذا التفاضل الأول تبعا لمفكوك تايلور للدالة f(x) عند النقطة x كما يلى:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots \qquad (٨-٢)$$

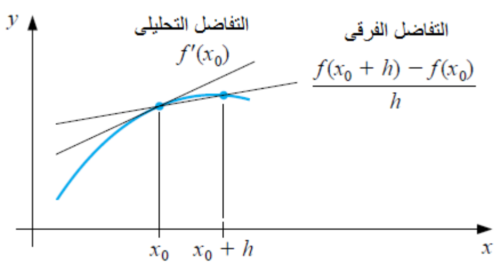
حيث $f^{(4)}(x)$ هى التفاضل الرابع للدالة f(x) و f''(x) هى تفاضلها الثالث وهكذا. من المعادلة (٨- ٢) يمكن كتابة التفاضل الأول كما يلى:

$$f'(x) = \frac{f(x+h)-f(x)}{h} - \frac{h}{2!}f''(x) - \frac{h^2}{3!}f'''(x) + \cdots ،$$

وهذه يمكن كتابتها على الصورة:

$$f'(x) = \frac{f(x+h)-f(x)}{h} - O(h) \qquad (٨-٣)$$

حيث O(h) هو مقدار خطأ يتناسب مع h. حيث أن هذا التفاضل يعتمد على قيمة الدالة عند النقطة x و النقطة x+h فإن التفاضل الأول الذى فى المعادلة (٥ -١) يسمى تفاضل الفرق الأمامى، أو تفاضل الفرق الأمامى أحادى الجانب. بنفس الطريقة يمكن كتابة تفاضل الفرق الخلفى على الصورة التالية حيث أنه يعتمد على f(x) و f(x-h):

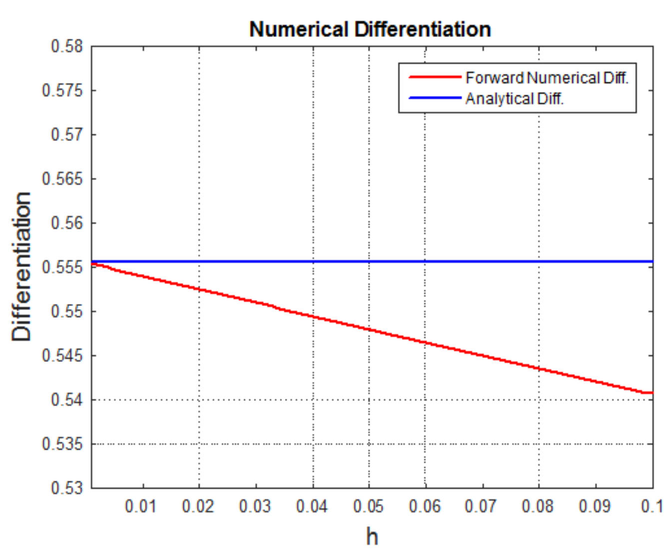


شكل ٨-١ التفاضل الفرقى والتفاضل التحليلى

$$f'(x) \approx \frac{f(x)-f(x-h)}{h} + O(h) \qquad (٨-٤)$$

شكل ٨-١ يوضح الفرق بين التفاضل الفرقى والتفاضل التحليلى، حيث التفاضل الفرقى هو الفرق بين نقطتين متقاربتين على الدالة مقسوما على المسافة بين النقطتين h، بينما التفاضل التحليلى هو تفاضل الصورة التحليلية للدالة والتعويض فى هذا التفاضل بقيمة المتغير x عند x_0.

مثال ٨-١ يوضح دقة هذا التفاضل.



شكل ٨-٢ الفرق بين التفاضل الفرقى الأمامى والتفاضل الحقيقى (التحليلى)

مثال ٨–١

إفترض الدالة f(x)=ln(x) والتى تفاضلها التحليلى هو f'(x)=1/x، عندما x=1.8 فإن f'(1.8)= 0.555556. شكل ٨–٢ يبين التفاضل الفرقى الأمامى تبعا للمعادلة (٨– ١) عند قيم مختلفة للمعامل h بدءا من h=0.001 حتى h=0.1 حيث نلاحظ أنه مع تقليل قيمة h فإن التفاضل الفرقى يقترب من التفاضل الحقيقى (التحليلى) حيث عندما h=0.1 فإن f'(x)= 0.5406722، وعندما h=0.001 فإن f'(x)= 0.55540129.

من الممكن كتابة صورة أكثر دقة لهذا التفاضل الأول تعتمد على الفرق المركزى حيث يعتمد هذا التفاضل على f(x+h) و f(x–h) كما يلى:

$$f'(x) \approx \frac{f(x+h)-f(x-h)}{2h} \qquad (٨–٥)$$

ويمكن إثبات أن هذه الصورة تكون أدق من الصورة التى فى المعادلة (٨– ١) حيث سنكتب مفكوك تايلور عند النقطة x على الصورتين التاليتين وكما فى المعادلة (٨– ٢):

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots$$

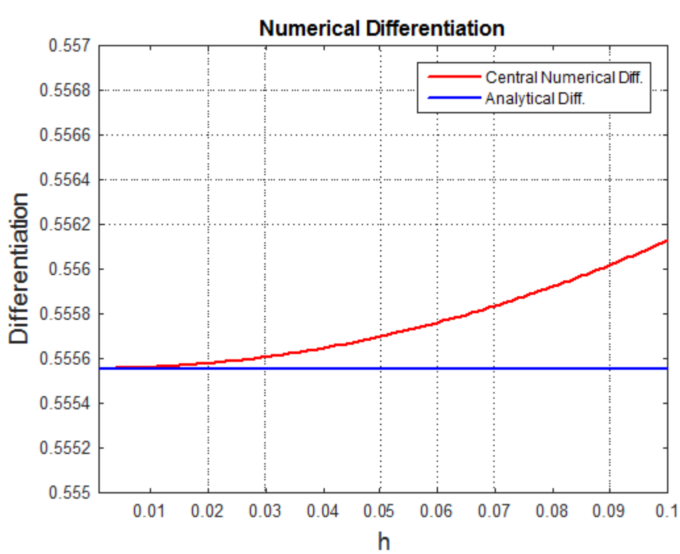
بطرح المعادلتين السابقتين نحصل على:

$$f'(x) = \frac{f(x+h)-f(x-h)}{2h} - \frac{h^2}{2x3!}f'''(x) - \cdots$$

وهذه المعادلة يمكن كتابتها على الصورة المدمجة التالية:

$$f'(x) \approx \frac{f(x+h)-f(x-h)}{2h} - O(h^2) \qquad (٨–٦)$$

حيث مقدار الخطأ هنا O(h2) يكون أصغر من الخطأ فى تقريب المعادلة (٨– ١) لأن h أصغر من الواحد كما ذكرنا، والخطأ فى هذه المعادلة يقال عنه أنه خطأ من الدرجة الثانية بينما الخطأ فى المعادلة (٨– ١) فيقال عنه خطأ من الدرجة الأولى.



شكل ٨–٣ الفرق بين التفاضل الفرقى المركزى والتفاضل الحقيقى

مثال ٨–٢

أعد مثال ٨–١ مستخدما التفاضل بالفرق المركزى. تذكر أن التفاضل عند x=1.8 يساوى f'(x)= 0.555556. شكل ٨– ٣ يبين التفاضل الفرقى المركزى تبعا للمعادلة (٨– ٥) عند قيم مختلفة للمعامل h بدءا من h=0.001 حتى h=0.1 حيث نلاحظ أنه مع تقليل قيمة h فإن التفاضل المركزى يقترب من التفاضل الحقيقى (التحليلى) حيث عندما h=0.1 فإن f'(x)= 0.00556128، وعندما h=0.001 فإن f'(x)= 5.555556e-07، حيث نلاحظ أن التفاضل المركزى أكثر دقة بكثير من التفاضل الأمامى.

يمكن كتابة التفاضل الثانى للدالة f(x) من مفكوك تايلور الأصلى كما يلى:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots$$

بجمع المعادلتين السابقتين نحصل على:

$$f''(x) = \frac{f(x+h)-2f(x)+f(x-h)}{h^2} + \frac{h^2}{12}f^{(4)}(x) + \cdots$$

والتى يمكن كتابتها كما يلى:

$$f''(x) \approx \frac{f(x+h)-2f(x)+f(x-h)}{h^2} + O(h^2) \qquad (٨–٧)$$

بنفس الطريقة يمكن إيجاد التفاضلات الأعلى مثل التفاضل الثالث كما يلى:

$$f'''(x) = \frac{f(x+2h)-2f(x+h)+2f(x-h)-f(x-2h)}{2h^3} + O(h^2) \qquad (٨–٨)$$

والتفاضل الرابع كما يلى:

$$f^{(4)}(x) = \frac{f(x+2h)-4f(x+h)+6f(x)-4f(x-h)+f(x-2h)}{h^4} + O(h^2) \qquad (٨–٩)$$

لاحظ أن هذه التفاضلات من النوع المركزى أو ثنائة الجانب. على الرغم من أن التفاضلات أحادية الجانب سواء الأمامية أو الخلفية التى فى المعادلتين (٨– ١) و (٨– ٤) يكون الخطأ فيهما أسوأ إلا أن التفاضلات أحادية الجانب تستخدم عادة عند نقاط الأطراف x_n و x_1.

٨–٣ التفاضل عن طريق الاستيفاء

إذا كانت نقاط البيانات للدالة المعطاة ليست متساوية فى المسافة فيما بينها، فإن تفاضل هذه الدالة بالطرق الفرقية السابقة لن تكون مناسبة وفى هذه الحالة يمكن استخدام طريقة الاستيفاء حيث يتم تقريب هذه النقاط بدالة استيفاء معينة ثم نحسب تفاضل دالة الاستيفاء ليكون هو التفاضل التقريبى للدالة المعطاة.

الفكرة هنا هى أنه إذا كان عدد نقاط البيانات هو n فإنه يمكننا إيجاد دالة من الدرجة n-1 تستوفى هذه البيانات، وهذه الدالة يمكن أن تكون على الصورة:

$$P(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots + a_1 x + a_0 \qquad (٨-١٠)$$

ثم نقوم بتفاضل المعادلة (٨- ١٠) لتمثل تفاضل مجموعة النقاط المعطاة. ونذكر هنا أننا رأينا العديد من طرق الاستيفاء فى الفصول السابقة ولقد رأينا أنه يجب أن تكون درجة دالة الاستيفاء أقل ما يمكن حتى نتجنب تذبذب دالة الاستيفاء بالذات عند أطراف البيانات المعطاة حيث أن عملية التفاضل من الممكن أن تزيد من هذا التذبذب.

باستخدام طريقة لاجرانج للاستيفاء فى الفصل ٥ كتبنا دالة استيفاء مجموعة من النقاط على الصورة التالية بإهمال الخطأ:

$$L(x) = \sum_{i=0}^{n} y_i r_i(x) \qquad (٨-١١)$$

بتفاضل هذه الدالة عند أى نقطة $x=x_k$ نحصل على:

$$L'(x) = \sum_{i=0}^{n} y_i {r'}_i(x_k) \qquad (٨-١٢)$$

مثال ٨-٣

بفرض n=1 والتعويض فى المعادلة (٨- ١١) نحصل على معادلة لاجرانج من الدرجة الأولى كما يلى:

$$L(x) = y_0 r_0(x) + y_1 r_1(x) \qquad (٨-١٣)$$

وهذه يمكن كتابتها على الصورة التالية بالتعويض عن كل من $r_0(x)$ و $r_1(x)$:

$$L(x) = y_0 \frac{x-x_1}{x_0-x_1} + y_1 \frac{x-x_0}{x_1-x_0} \qquad (٨-١٤)$$

بإجراء التفاضل على المعادلة (٨ -١٤) نحصل على:

$$L'(x) = -y_0 \frac{1}{x_1-x_0} + y_1 \frac{1}{x_1-x_0}$$

وهذه يمكن كتابتها على الصورة التالية:

$$L'(x) = \frac{y_1-y_0}{x_1-x_0} \qquad (٨-١٥)$$

كما نعلم فإن $y_1=f(x_1)$ و $y_0=f(x_0)$، وبفرض أن $x_1=x_0+h$ فإن المعادلة (٨- ١٥) يمكن كتابتها كالتالى:

$$L'(x) = \frac{f(x_0+h)-f(x_0)}{h} \qquad (٨-١٦)$$

وهى نفس معادلة التفاضل الفرقى التى حصلنا عليها فى المعادلة (٨- ١) . تذكر بأنه تم إهمال خطأ تقريب لاجرانج فى كل المعادلات السابقة لتبسيط الأمور.

٨–٤ التفاضل فى ماتلاب

يوجد فى ماتلاب دالة جاهزة البناء لحساب الفرق بين كل نقطتين متتاليتين على أى مجموعة بيانات مدخلة. هذه الدالة هى:

y=diff(x)

حيث تقوم هذه الدالة بحساب الفرق بين كل نقطتين متتاليتين فى المتجه x. فإذا كان المتجه x يتكون من عدد n من العناصر x_i حيث i تتغير من 1 إلى n، كالتالى: $x_1, x_2, \ldots, x_n$ فإن المتجه y سيتكون من عدد n−1 من العناصر كالتالى: $y_1=x_1-x_2$ و $y_2=x_2-x_3$ و $y_3=y_4-y_3$ وهكذا إلى $y_{n-1}=y_{n-1}-y_n$. كمثال على ذلك سنحسب الفروق بين مربعات الأرقام من ١ إلى ١٠ كما يلى:

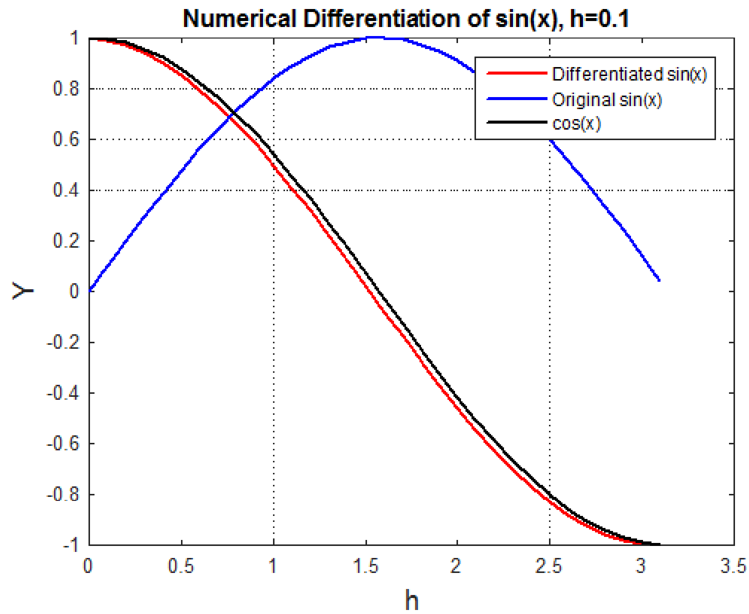
```
>> x=(1:10).^2
x =
   1    4    9   16   25   36   49   64   81   100
>> y=diff(x)
y =
   3    5    7    9   11   13   15   17   19
```

مثال ٨–٤

يمكن توضيح ذلك أكثر بمثال نقوم فيه بتفاضل الدالة f(x)=sin(x) عن طريق تقسيم الدالة f(x) إلى نقاط متساوية البعد بين كل منهما هو h=0.1، ثم نجرى دالة الفرق ونقسم هذه الفروق على المسافة بين كل نقطتين (h) فنحصل على تفاضل الدالة f(x)، ثم نقارن التفاضل الرقمى الناتج مع التفاضل التحليلى للدالة f(x)=sin(x) وهو cos(x) عن طريق رسم كل منهما مع x كما فى شكل ٨–٤. البرنامج التالى يقوم بهذه العملية:

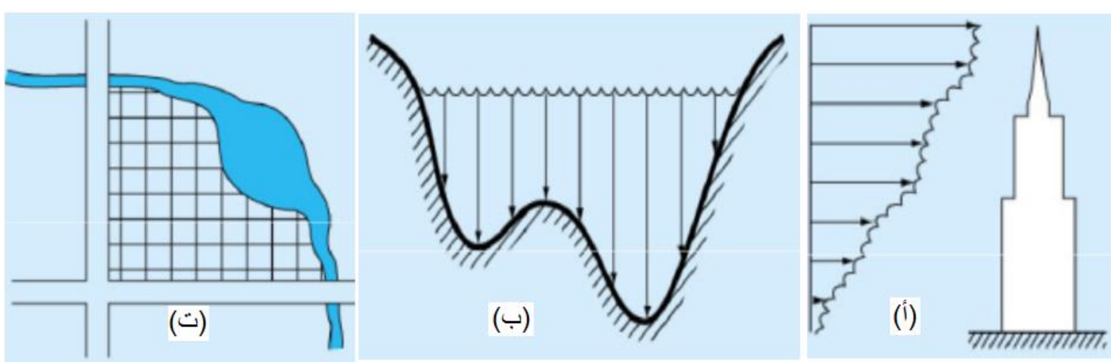
```
% Numerical Diff. of a sin(x) using the diff(x) function
h = 0.1;
x = 0:h:pi;
n=size(x);
m=size(x,2)-1;%to make y the same size as x
y=diff(sin(x))/h;
plot(x(1,1:m),y,'r', 'linewidth', 1.5);
hold on; grid;
y1=sin(x);
y2=cos(x);
plot(x,y1 ,'b', 'linewidth', 1.5);hold on;
plot(x,y2 ,'k', 'linewidth', 1.5);
title('Numerical Differentiation of sin(x), h=0.1','fontsize',12)
ylabel('Y','fontsize',14)
xlabel('h','fontsize',14)
legend('Differentiated sin(x)','Original sin(x)', 'cos(x)')
```

لاحظ أنه بتقليل المسافة بين كل نقطتين h سيكون منحنى التفاضل التقريبي (الأحمر) أكثر انطباقا على منحنى التفاضل الحقيقى (الأسود).



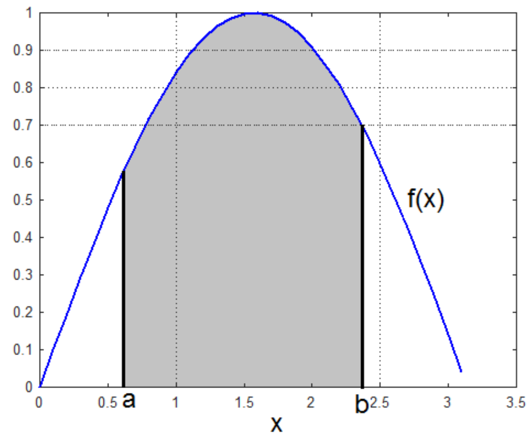
شكل ٨-٤ استخدام دالة ماتلاب diff(x) لتفاضل الدالة f(x)=sin(x) والمقارنة بتفاضلها التحليلى cos(x)

التكامل العددى Numerical Integration

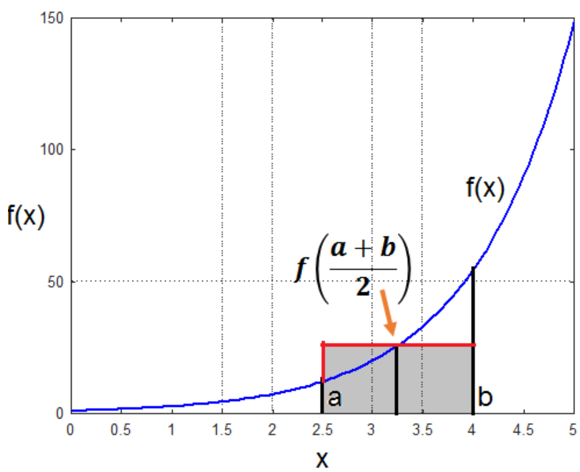


شكل ٨-٥ إستخدام التكامل العددى فى (أ) فى حساب مجموع قوى الرياح المؤثرة على ناطحة سحاب، (ب) حساب مساحة مقطع غير منتظم الشكل، (ت) حساب مساحة قطعة أرض غير منتظمة الشكل.

التكامل العددى بالتأكيد هو الأكثر شيوعا والأكثر استخداما نتيجة استخدامة فى الحياة العملية فى الكثير من التطبيقات. شكل ٨-٥ يبين ثلاثة من هذه التطبيقات حيث فى شكل ٨-٥أ يتم استخدام التكامل العددى لحساب مجموع القوى المؤثرة على ناطحة سحاب نتيجة الرياح المؤثرة عليها من قاعدتها حتى قمتها حيث يكون تأثير هذه القوة أكبر ما يمكن عند قمة الناطحة. فى شكل ٨-٥ب يمكن استخدام التكامل العددى أيضا فى حساب مساحة مقطع أحد الأشكال التى من الصعب حسابها باستخدام طرق التكامل التحليلية. فى شكل ٨-٥ت يمكن أيضا استخدام التكامل العددى فى حساب مساحة قطعة أرض غير منتظمة الحدود كما فى الشكل، وأيضا من الصعب جدا استخدام الطرق التحليلية فى هذا التكامل. معنى ذلك أنه من المنطقى أن يكون استخدام التكامل العددى فى التطبيقات التى يصعب فيها الحصول على تكامل تحليلى، مما يعنى أن التكامل الذى سنحصل عليه بالطرق العددية سيكون تقريبى وستكون به نسبة خطأ تختلف من طريقة لأخرى. فى العادة يطلق على عملية التكامل العددى "التربيع quadrature" حيث جرت العادة فى حسابات المساحات أن يتم تقسيمها إلى مربعات صغيرة يتم عددها لحساب المساحة الكلية. من ذلك نرى أيضا أن التكامل العددى لابد أن يكون تكاملا محدودا فى منطقة محددة لمتغير التكامل كما فى المعادلة (٨- ١٧) حيث فى هذه المعادلة يتم إجراء التكامل فى المنطقة من x=a حتى x=b، وستكون قيمة هذا التكامل المحدود هى المساحة الواقعة تحت المنحنى f(x) فى المنطقة المحددة [a b] كما فى شكل ٨-٦.



شكل ٨-٦ التكامل العددى يكون على مدى محدد للمتغير x فى المدى [a b] مثلا.



شكل ٨-٧ التكامل بطريقة المستطيل أو النقطة المتوسطة

$$shaded\ area = \int_a^b f(x)dx \qquad (٨-١٧)$$

لاحظ أن الدالة f(x) المطلوب تكاملها فى المدى [a b] من الممكن أن تكون معلومة إما فى صورة تحليلية أو فى صورة نقاط محددة عند x_i حيث i تتغير فى المدى [a b].

هناك العديد من طرق التحليل العددى وكلها تقريبا تقرب الدالة f(x) إلى كثيرة حدود فى المدى [a b] باستخدام أحد طرق الاستيفاء التى تم دراستها مسبقا. بالطبع يجب استخدام درجات منخفضة لدالة الاستيفاء حتى نتجنب التذبذب بين نقاط الاستيفاء أو العقد. سنذكر هنا أكثر هذه الطرق شيوعا واستخداما.

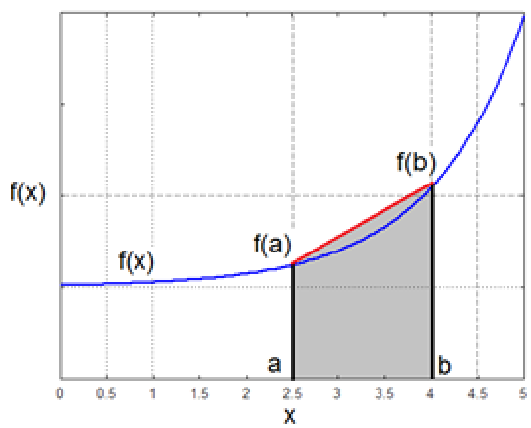
٨–٥ طريقة النقطة المتوسطة midpoint للتكامل

فى هذه الطريقة يتم تقريب أو استيفاء الدالة f(x) أو مجموعة النقاط فى المدى [a b] بخط أفقى يمر بالدالة f(x) عند نقطة المنتصف بين النقطتين a و b بحيث يتم تقريب المساحة تحت المنحنى بمستطيل عرضه المسافة [a b] وارتفاعه هو قيمة الدالة عند نقطة المنتصف $f\left(\frac{a+b}{b}\right)$، كما فى شكل ٨–٧. هذا النوع من الاستيفاء يسمى استيفاء من الدرجة صفر. فى هذه الحالة يمكن كتابة قيمة التكامل كما فى المعادلة (٨– ١٨) كما يلى:

$$\int_a^b f(x)\,dx \approx (b-a)f\left(\frac{a+b}{2}\right) \qquad (١٨–٨)$$

هذه الطريقة تسمى طريقة المستطيل أو النقطة المتوسطة للتكامل العددى.

٨–٦ طريقة شبه المنحرف trapezoid للتكامل



شكل ٨–٨ التكامل العددى بقانون شبه المنحرف

فى هذه الطريقة يتم استيفاء الدالة f(x) بين النقطتين a و b بخط مستقيم يمر بقيمتى الدالة عند هاتين النقطتين f(a) و f(b) كما هو موضح فى شكل ٨–٨. لذلك فإن الاستيفاء هنا يكون من الدرجة الأولى. هذا الخط المستقيم الناتج يكون شبه منحرف قاعدته هى h=b-a وجانبيه هما f(a) و f(b) ولذلك يمكن كتابة مساحة شبه المنحرف الناتج حاصل ضرب القاعدة فى متوسط الجانبين كما فى المعادلة (٨– ١٩) التالية:

$$\int_a^b f(x)\,dx \approx (b-a)\left[\frac{f(b)-f(a)}{2}\right] \qquad (١٩–٨)$$

لاحظ أننا نستخدم علامة التساوى التقريبى (≈) فى كل من المعادلتين (٨– ١٨) و (٨– ١٩) مما يعنى أن هناك نسبة خطأ فى هذا التكامل وأنه لا يمكن أن يساوى تماما التكامل الحقيقى للدالة أو التكامل التحليلى.

٨–٧ طريقة شبه المنحرف المركب composite trapezoid للتكامل

لزيادة دقة التكامل بطريقة شبه المنحرف يمكن تقسيم المدى [a b] إلى عدد n من المقاطع وحساب مساحة شبه المنحرف الممثل لكل مقطع كما فى شكل ٨–٩. بفرض أن $a=x_1<x_2<x_3\ldots<x_{n+1}=b$ فإنه يمكن كتابة المساحة الواقعة تحت المنحنى فى المدى [a b] كما فى المعادلة (٨– ٢٠) كما يلى:

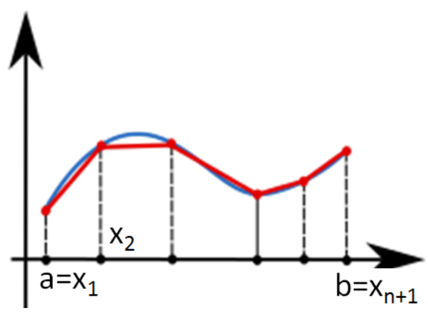
$$\int_a^b f(x)\,dx \approx \frac{h}{2}\sum_{k=1}^{n}(f(x_{k+1})+f(x_k)) \qquad (٨-٢٠)$$

حيث h=(b-a)/n. بفك المجموع فى المعادلة (٨– ٢٠) يمكن إعادة كتابة هذه المعادلة كما يلى:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{2n}(f(x_1)+2f(x_2)+2f(x_3)+\cdots+f(x_{n+1})) \qquad (٨-٢١)$$

المعادلة (٨– ٢١) تسمى معادلة حساب التكامل العددى بطريقة شبه المنحرف المركب.

٨–٨ التكامل بشبه المنحرف فى ماتلاب



شكل ٨–٩ التكامل العددى باستخدام قانون شبه المنحرف المركب

يحتوى ماتلاب على دالتين جاهزتى البناء لحساب التكامل العددى باستخدام طريقة شبه المنحرف، والدالتان هما:

الدالة الأولى: z=trapz(y)

الدالة الثانية: z=trapz(x,y)

الدالة الأولى تحسب التكامل التقريبى للمتغير y باستخدام طريقة شبه المنحرف مع اعتبار التباعد بين النقطتين أو قاعدة شبه المنحرف h تساوى واحد، ويضع الناتج فى z. إذا كانت h تختلف عن الواحد فإنه يتم ضربها فى z.

الدالة الثانية تحسب التكامل التقريبى للمتغير y بالنسبة للمتغير x باستخدام طريقة شبه المنحرف. كل من x و y من الممكن أن يكونا متجهين بشرط أن يكون لهما نفس الطول، وسنوضح ذلك بالأمثلة.

مثال ٨–٥

إحسب التكامل التالى مستخدما طريقة شبه المنحرف فى الماتلاب: $I=\int_0^{\pi}\sin(x)\,dx$

أولا الحل التحليلى لهذه الدالة هو:

$$I=\int_0^{\pi}\sin(x)\,dx=\cos(x)|_0^{\pi}=\cos(0)-\cos(\pi)=2$$

ثانيا الحل باستخدام الدالة z=trapz(y) سيكون كما يلى بعد تقسيم المدى من x=0 إلى x=π إلى عدد اختيارى من المقاطع ولتكن مائة قسم:

```
>> X = 0:pi/100:pi;
>> Y = sin(X);
>> Z = pi/100*trapz(Y)
Z =
   1.9998
```

حيث كما ذكرنا أن الدالة trapz(y) تعتبر أن عرض المقطع يساوى 1 لذلك لزم الضرب فى عرض المقطع وهو pi/100.

ثالثا باستخدام الدالة z=trapz(x,y) سيكون كما يلى:

```
>> X = 0:pi/100:pi;
>>Y = sin(X);
>> Z = trapz(X,Y)
Z =
   1.9998
```

وهى نفس النتيجة السابقة. لاحظ أن الخطأ فى استخدام كل من الدالتين يساوى 0,0002.

مثال ٨–٦

الجدول التالى يبين العلاقة بين الزمن والسرعة لجسم متحرك. إحسب المسافة المقطوعة فى الزمن من 0 حتى 3 وحدات زمنية.

الزمن (ثانية)	0.0	1.0	2.0	3.0
السرعة (متر/الثانية)	0.0	10	12	14

فى هذه الحالة ستكون المسافة هى تكامل السرعة فى الفترة من 0 إلى 3 كما يلى:

$$distance = \int_0^3 v\,dt$$

أولا الحل العددى باستخدام قانون شبه المنحرف المركب الذى فى المعادلة (٨– ٢١) حيث الفترة الزمنية من 0 إلى 3 مقسمة إلى ثلاث مقاطع (صفر حتى ١ و ١ حتى ٢ و ٢ حتى ٣):

$$D = \frac{b-a}{2n}(f(x_1) + 2f(x_2) + 2f(x_3) + \cdots + 2f(x_{n+1}))$$
$$D = \frac{3-0}{2x3}(f(0) + 2f(1) + 2f(2) + f(3))$$
$$D = \frac{3-0}{2x3}(0 + 2x10 + 2x12 + 14)$$
$$D = 0.5(20 + 12 + 14) = 29$$

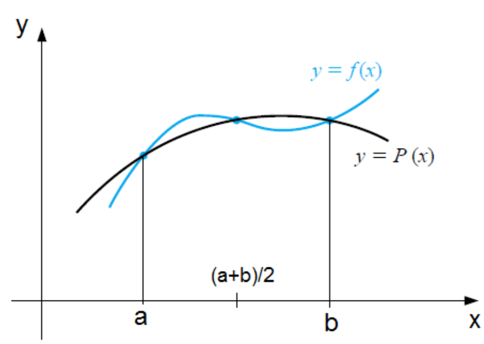
ثانيا باستخدام دالة ماتلاب z=trapz(x,y):

```
>> x=0:3
x =
   0   1   2   3
>> y=[0 10 12 14]
y =
   0   10   12   14
>> D=trapz(x,y)
D =
   29
```

وهى نفس الإجابة التى حصلنا عليها بالتعويض المباشر فى قانون التكامل بشبه المنحرف المركب.

٨-٩ قانون سمبسون Simpson للتكامل العددى

لقد رأينا أن طريقة المستطيل أو النقطة المتوسطة كانت عبارة عن استيفاء للدالة f(x) من الدرجة صفر، وطريقة شبه المنحرف كانت عبارة عن استيفاء من الدرجة الأولى، وبالتالى كما سنرى فإن طريقة سمبسون هى استيفاء من الدرجة الثانية لثلاث نقاط على هذه الدالة هم النقطة a والنقطة b ونقطة المنتصف التى سنسميها m=(a+b)/2 حيث a و b هما المدى الذى يتم فيه حساب التكامل كما فى شكل ٨-١٠.



شكل ٨-١٠ طريقة سمبسون للتكامل العددى

استيفاء لاجرانج لثلاث نقاط يمكن كتابته كما يلى وكما رأينا فى الفصل ٥،

$$p(x) = \sum_1^3 f(x_i) r_i(x) \qquad (٨-٢٢)$$

حيث $r_i(x)$ هى دوال القاعدة للجرانج كما رأينا مسبقا، و f(x) هى الدالة المطلوب تكاملها. بتكامل الدالة p(x) نحصل على:

$$I = \int_a^b p(x)\,dx = \sum_1^3 \left[f(x_i) \int_a^b r_i(x)\,dx \right] \qquad (٨-٢٣)$$

بالتعويض عن دوال القاعدة r_i فى المعادلة (٨- ٢٣) نحصل على:

$$I = f(a) \int_a^b \frac{(x-m)(x-b)}{(a-m)(a-b)} dx + f(m) \int_a^b \frac{(x-a)(x-b)}{(m-a)(m-b)} dx + f(b) \int_a^b \frac{(x-a)(x-m)}{(b-a)(b-m)} dx \qquad (٨-٢٤)$$

بإجراء هذه التكاملات وبعد الكثير من الاختصارات يمكن الوصول إلى الصورة النهائية لهذا التكامل كما يلى:

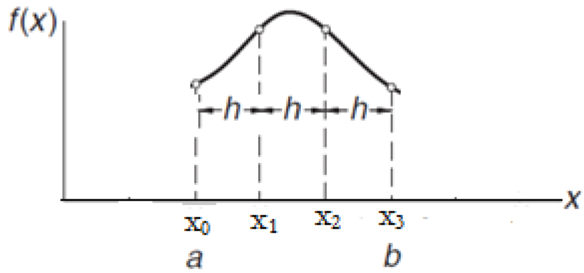
$$I=\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left[f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right] \qquad (٨-٢٥)$$

فى المعادلة السابقة بوضع h=(b-a)/2 فإن المعادلة السابقة يمكن كتابتها كما يلى:

$$I=\int_a^b f(x)\,dx \approx \frac{h}{3}\left[f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right] \qquad (٨-٢٦)$$

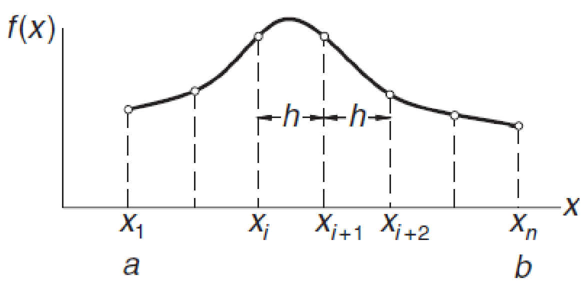
هذا القانون (المعادلة (٨- ٢٦) يسمى قانون 1/3 للتكامل بطريقة سمبسون حيث كما رأينا يتم استخدام ثلاث نقاط على الدالة المراد تكاملها عدديا للحصول على استيفاء لاجرانج من الدرجة الثانية.

x_0 x_1 x_2 x_3



شكل ٨-١١ التكامل العددى بالقانون 3/8 لسمبسون

هناك طريقة ثانية للتكامل بطريقة سمبسون وهى تستخدم بكثرة أيضا حيث أنها أكثر دقة وهذه الطريقة تسمى قانون 3/8 للتكامل بطريقة سمبسون. فى هذه الطريقة يتم استخدام أربع نقاط فى المدى [a b] كما فى شكل ٨-١١ للحصول على استيفاء لاجرانج من الدرجة الثالثة. فى هذه الحالة يمكن كتابة استيفاء لاجرانج لأربع نقاط بنفس طريقة المعادلات (٨- ٢٢) و (٨- ٢٣) و (٨- ٢٤) ، وبعد إجراء التكاملات على دوال قاعدة لاجرانج والاختصارات نحصل على القانون النهائى للتكامل كما يلى:



شكل ٨-١٢ التكامل العددى بقانون سمبسون المركب

$$I=\int_a^b f(x)\,dx \approx \frac{3h}{8}\left[f(x_0)+3f(x_1)+3f(x_2)+f(x_3)\right] \qquad (٨-٢٧)$$

حيث h=(b-a)/3 فى هذه الحالة، و $x_0=a$ و $x_3=b$ ، و x_1 هى نقطة التقسيم الثانية بجوار النقطة a و x_2 هى نقطة التقسيم الثالثة قبل النقطة b. هذا القانون يعرف بالقانون 3/8 لسمبسون كما ذكرنا.

٨-١٠ قانون سمبسون المركب للتكامل العددى

لزيادة دقة التكامل يمكن تقسيم مدى التكامل إلى مقاطع كما فعلنا مع قانون شبه المنحرف المركب وتجميع هذه التكاملات. شكل ٨-١٢ يبين المدى [a b] وقد تم تقسيمه إلى مقاطع كل منها عرضه h يساوى h=(b-a)/n-1 حيث n هى عدد نقاط التقسيم كما فى الشكل. بتطبيق قانون 1/3 سمبسون الذى فى المعادلة (٨- ٢٥) على النقاط الثلاث x_i و x_{i+1} و x_{i+2} يمكن كتابة التكامل التالى:

$$\int_{x_i}^{x_{i+2}} f(x)\,dx = \frac{h}{3}[f(x_i) + 4f(x_{i+1}) + f(x_{i+2})] \qquad (٨-٢٨)$$

بالتعويض من المعادلة (٨- ٢٨) فى المعادلة (٨- ٢٥) على كل ثلاث نقط بدءا من x_1 حتى x_n يمكن كتابة ما يلى:

$$\int_a^b f(x)\,dx = \int_{x_1}^{x_n} f(x)\,dx = \sum_{i=1.3.5....}^{n-2} \left[\int_{x_i}^{x_{1+2}} f(x)\,dx\right]$$

بالتعويض فى المعادلة السابقة بقيمة التكامل من المعادلة (٨- ٢٨) وبإجراء بعض الاختصارات يمكننا كتابة التكامل النهائى كما يلى:

$$I = \int_a^b f(x)\,dx \approx \frac{h}{3}[f(x_1) + 4f(x_2) + 2f(x_3) + 4f(x_4) + \cdots + 4f(x_{n-1}) + f(x_n)] \quad (٨-٢٩)$$

لاحظ أن المعادلة (٨- ٢٩) تفترض أن عدد نقاط التقسيم n تكون عدد فردى لكى يكون هناك عدد زوجى من المقاطع. المعادلة (٨- ٢٩) يمكن كتابتها فى الصورة المدمجة التالية:

$$I = \int_a^b f(x)\,dx \approx \frac{h}{3}\sum_{i=1}^{(n-1)/2}[f(x_{2i-1}) + 4f(x_{2i}) + f(x_{2i+1})] \qquad (٨-٣٠)$$

إذا كان هناك عدد زوجى من نقاط التقسيم n، أى أن هناك عدد فردى من المقاطع، فإنه فى هذه الحالة يمكن تطبيق القانون 3/8 لسمبسون على أول أربع نقاط، ثم تقسيم القانون 1/3 على النقاط المتبقية.

مثال ٨–٧

مطلوب حساب تكامل الدالة f(x)=ex فى المدى x=0 إلى x=2 باستخدام قوانين سمبسون.

أولا القيمة الحقيقية لهذا التكامل يمكن كتابتها كالتالى:

$$\int_0^2 e^x\,dx = e^x|_0^2 = e^2 - e^1 = 6.389056098930650$$

ثانيا تكامل عددى بقانون سمبسون باعتبار المدى [0 2] مقطع واحد:

$$\int_0^2 e^x\,dx \approx \frac{2-0}{6}[e^0 + 4e^1 + e^2] = 6.4207278$$

ثالثا تكامل عددى باستخدم قانون سمبسون مع تقسيم المدى [0 2] إلى مقطعين [0 1] و [1 2]:

$$\int_0^2 e^x\,dx \approx \int_0^1 e^x\,dx + \int_1^2 e^x\,dx \approx \frac{1-0}{6}[e^0 + 4e^{0.5} + e^1] + \frac{2-1}{6}[e^1 + 4e^{1.5} + e^2] \approx 6.39121018$$

كما نلاحظ فإنها أقرب كثيرا من تكامل المقطع الواحد.

رابعا تكامل عددى باستخدم قانون سمبسون مع تقسيم المدى [0 2] إلى أربع مقاطع [0 0.5] و [0.5 1] و [1 1.5] و [1.5 2] حيث سيكون التكامل كما يلى:

$$\int_0^2 e^x\,dx \approx \int_0^{0.5} e^x\,dx + \int_{0.5}^1 e^x\,dx + \int_1^{1.5} e^x\,dx + \int_{1.5}^2 e^x\,dx = 6.38919372$$

وهو بالطبع أدق من الطريقتين السابقتين وأقرب إلى التكامل الحقيقى. لاحظ أننا نستخدم القانون 1/3 فى كل مقطع من هذه المقاطع فى الحالات ثانيا وثالثا ورابعا.

مثال ٨–٨

إحسب تكامل الدالة f(x) المعطاة بالجدول التالى فى المدى من 0 حتى 2.5 مستخدما قوانين سمبسون.

x	0	0.5	1	1.5	2	2.5
f(x)	1.5000	2.0000	2.0000	1.6364	1.2500	0.9565

حيث أن عدد النقاط زوجى (ستة) وبالتالى فإن عدد المقاطع يكون فردى (خمسة) فإنه لا يمكن استخدام القانون 1/3 وحده، وبالتالى فإننا سنستخدم القانون 3/8 للنقاط الأربعة الأولى (0 و 0.5 و 1 و1.5)، والقانون 1/3 لآخر ثلاث نقاط كما يلى:

$$I=\frac{3x0.5}{8}[f(0)+3f(0.5)+3f(1)+f(1.5)]+\frac{0.5}{3}[f(1.5)+4f(2)+f(2.5)]$$
$$I=\frac{3x0.5}{8}[1.5+3x2+3x2+1.6364]+\frac{0.5}{3}[1.6364+4x1.25+0.9565]=4.1036$$

٨–١١ التكامل بقانون سمبسون فى ماتلاب

الدالة سابقة البناء فى ماتلاب والتى تقوم بإجراء التكامل العددى لأى دالة هى:

$$Q=quad('fun'.a.b)$$

حيث تقوم هذه الدالة بإجراء التكامل العددى بطريقة سمبسون على الدالة fun فى المدى [a b] بطريقة تكرارية حتى يكون الخطأ بين تكاملين متتاليين يساوى $1x10^{-6}$. نعنى بالطريقة التكرارية أنه يحسب التكامل على المدى بالكامل، ثم على المدى مقسم إلى قسمين، ثم ثلاثة أقسام، وهكذا إلى أن يصل إلى القيمة المطلوبة من الخطأ.

مثال ٨–٩

استخدام دالة ماتلاب quad( ) لحساب التكامل العددى $\int_0^2 e^x\,dx$.

سنكتب الدالة مباشرة فى مساحة عمل ماتلاب كما يلى:

```
>> quad('exp(x)',0,2)
ans =
   6.389056104485924
```

نتذكر من المثال ٨-٩ أن القيمة الحقيقية لهذا التكامل كانت: 6.38905609830650 وبالمقارنة نجد أن النتيجة مقربة فعلا لأقرب $1x10^{-6}$. صورة أخرى من صور الدالة quad( ) كما يلى:

```
>> [I,n]=quad('exp(x)',0,2)
I =
  6.389056104485924
n =
   29
```

حيث I هى قيمة التكامل و n هى عدد الخطوات التكرارية حتى يصل إلى النسبة المحددة للخطأ وهى $1x10^{-6}$.

يمكن تحديد نسبة الخطأ لأى نسبة أخرى غير النسبة التلقائية التى تساوى $1x10^{-6}$بالنص عليها كما يلى [I,n]=quad('fun',a,b,tol) حيث tol هو الخطأ المطلوب كما يلى:

```
>> [I,n]=quad('exp(x)',0,2,0.0001)
I =
  6.389056901245308
n =
   13
```

حيث تم النص هنا على أن نسبة الخطأ تساوى 0.0001، ولذلك وصل إلى النتيجة بعد ١٣ محاولة فقط بدلا من ٢٩ محاولة عندما كان الخطأ يساوى $1x10^{-6}$.

فى حالة أن تكون الدالة f(x) دالة مركبة ويصعب كتابتها مباشرة فى نفس الأمر quad( ) فإنه يمكن كتابتها فى ملف دالة function file والنداء عليها كما يلى حيث كتبنا نفس الدالة f(x) فى ملف دالة كما يلى:

```
function [ y ] = myfunction( x )
% to demonstrate using function in numerical integration
 y = exp(x);
end
```

ثم من مساحة عمل الماتلاب يمكن تنفيذ التكامل كما يلى:

```
>> [I,n]=quad(@myfunction,0,2)
I =
  6.389056104485924
n =
   29
```

وهى نفس النتيجة السابقة. يرجى مراجعة ملفات الدوال فى أى مرجع للماتلاب.

طريقة أخرى للتعبير عن الدالة f(x) فى الأمر quad( ) تكون باستخدام الأمر inline( ) الذى يعبر عن نص أى دالة كما يلى:

```
>> f=inline('exp(x)');
>> I=quad(f,0,2)
I =
  6.389056104485924
```

٨-١٢ تكامل رومبيرج Romberg

هذه ليست طريقة جديدة للتكامل ولكن يمكن النظر إليها على أنها خطوات عامة لتحسين دقة أى تقريب عندما يكون مسلك الخطأ المصاحب لهذا التقريب معروفا كما هو الحال فى التفاضل العددى والتكامل العددى. سنوضح ذلك بمثال نحول فيه دقة التكامل التقريبى من الدرجة الثانية إلى دقة تقريب من الدرجة الرابعة وسنبدأ ذلك بإعادة كتابة طريقة التكامل عن طريق شبه المنحرف المركب التى فى المعادلة (٨- ٢١) كما يلى:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{2n}\left(f(x_1)+2f(x_2)+2f(x_3)+\cdots+f(x_{n+1})\right) \qquad (٣١-٨)$$

بوضع h=(b-a)/n يمكن كتابة المعادلة (٨- ٣١) كما يلى:

$$\int_a^b f(x)\,dx = h\big(0.5f(x_1)+f(x_2)+f(x_3)+\cdots+0.5f(x_{n+1})\big)+E \qquad (٣٢-٨)$$

حيث تم استخدام علامة التساوى الحقيقى وإضافة الخطأ E. هذا الخطأ يتناسب مع قوى المقطع h بحيث يمكن كتابته على الصورة العامة التالية:

$$I = T(h)+e_1h^2+e_2h^4+\cdots+e_ih^i+\cdots \qquad (٣٣-٨)$$

حيث $I=\int_a^b f(x)\,dx$ و $T(h)=h\big(0.5f(x_1)+f(x_2)+f(x_3)+\cdots+0.5f(x_{n+1})\big)$، و وباقى المقادير تمثل الخطأ E. المقادير ei عبارة عن ثوابت لا تعتمد على عرض المقطع h. باستخدام نصف العدد من النقاط تصبح h الجديدة تساوى 2h، وبالتعويض عنها فى المعادلة (٨- ٣٣) نحصل على:

$$I = T(2h)+e_1(2h)^2+e_2(2h)^4+\cdots+e_i(2h)^i+\cdots$$

$$I = T(2h)+4e_1h^2+16e_2h^4+\cdots+e_i(2h)^i+\cdots \qquad (٣٤-٨)$$

للتخلص من مقدار الخطأ الذى من الدرجة الثانية فإننا سنضرب المعادلة (٨- ٣٣) فى ٤ ونطرح منها المعادلة (٨- ٣٤) وبالتالى سنحصل على ما يلى:

$$3I = 4T(h)-T(2h)-12e_2h^4+\cdots$$

ومنها يمكن كتابة التكامل I على الصورة التالية:

$$I=\frac{4T(h)-T(2h)}{3}+E$$

حيث E هو مقدار الخطأ الذى يتناسب مع h4 هذه المرة، وهو بالطبع أصغر بكثير من الخطأ الذى فى معادلة شبه المنحرف الأساسية. المهم الآن هو القيمة التقريبية الجديدة للتكامل والتى يمكن كتابتها كما يلى:

$$I \approx \frac{4T(h)-T(2h)}{3}$$

لاحظ استخدام علامة التساوى التقريبى (≈) مرة أخرى لإهمال الخطأ. بالتعويض عن كل من T(h) و T(2h) من المعادلة (٨- ٣٣) يمكن كتابة التكامل السابق كما يلى:

$$I \approx \frac{h}{3}\Big[\big(2f(x_1)+4f(x_2)+4f(x_3)+\cdots+2f(x_{n+1})\big)-\big(f(x_1)+2f(x_3)+ 2f(x_5)+\cdots+f(x_{n+1})\big)\Big]$$

بتجميع المقادير المتشابهة فى المعادلة السابقة يمكن كتابة القيمة التقريبية للتكامل كما يلى:

$$I \approx \frac{h}{3}\big(f(x_1)+4f(x_2)+2f(x_3)+4f(x_4)+2f(x_5)\ldots+f(x_{n+1})\big) \qquad (٨-٣٥)$$

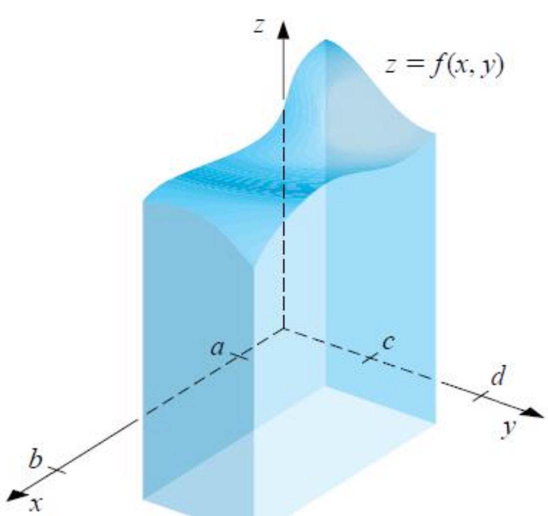
وهى نفس معادلة حساب التكامل المركب باستخدام قانون سمبسون كما فى المعادلة (٨- ٢٩).

٨-١٣ التكامل العددى ثنائى الأبعاد

كل الطرق التى تم شرحها مع التكامل العددى أحادى البعد، أى فى متغير واحد، يمكن استخدامها لإجراء التكامل العددى ثنائى الأبعاد، بل ومتعدد الأبعاد. إفترض أننا نريد حساب التكامل $\int_a^b \int_c^d f(x.y)\,dxdy$ حيث x تتغير فى المدى من a حتى b و y تتغير فى المدى من c حتى d. سنستخدم قانون سمبسون المركب لحساب هذا التكامل مع العلم أنه يمكن استخدام أى طريقة أخرى. لاحظ أن الدالة f(x,y) تمثل سطح وحساب هذا التكامل الثنائى يعنى حساب حجم المجسم القائم فوق المساحة المحددة بالنقاط a و b و c و d كما فى شكل ٨-١٣. هذا التكامل ثنائى الأبعاد يمكن كتابته على الصورة التالية:

$$\int_a^b \int_c^d f(x.y)\,dxdy = \int_a^b \left(\int_c^d f(x.y)\,dy\right)dx \qquad (٨-٣٦)$$

سنستخدم قانون سمبسون للتكامل المركب لحساب التكامل الداخلى $\int_c^d f(x.y)\,dy$ حيث فى هذه الحالة يعتبر المتغير x ثابت بالنسبة للمتغير y. باستخدام عدد زوجى من النقاط n لتقسيم المدى [a b] إلى مقاطع كل منها يساوى h=(b-a)/n، وأيضا عدد زوجى آخر



شكل ٨-١٣ التكامل العددى ثنائى الأبعاد

m من النقاط لتقسيم المدى [c d] إلى مقاطع كل منها يساوى k=(d-c)/m. بذلك يمكن إعادة كتابة قانون سمبسون، المعادلة (٨- ٢٩) ، للعدد m من النقاط فى الصورة المدمجة التالية:

$$\int_c^d f(x.y)dy \approx \frac{k}{3}\left[f(x.y_0)+2\sum_{j=1}^{\left(\frac{m}{2}\right)-1} f(x.y_{2j})+4\sum_{j=1}^{m/2} f(x.y_{2j-1})+f(x.y_m)\right] \qquad (٨\text{-}٣٧)$$

لاحظ أنه تم إهمال الكمية الخاصة بالخطأ فى المعاجلة (٨- ٣٧) ولذلك استخدمنا علامة التساوى التقريبى (≈).

لاحظ أيضا ترقيم النقاط من y_0 حتى y_m بدلا من y_1 حتى y_{m+1} كما فى المعادلة (٨- ٢٩).

بالتعويض بالتكامل الذى فى المعادلة (٨- ٣٧) فى المعادلة (٨- ٣٦) نحصل على ما يلى:

$$\int_a^b\int_c^d f(x.y)\,dxdy \approx \int_a^b\left(\frac{k}{3}\left[f(x.y_0)+2\sum_{j=1}^{\left(\frac{m}{2}\right)-1} f(x.y_{2j})+4\sum_{j=1}^{m/2} f(x.y_{2j-1})+ f(x.y_m)\right]\right)dx \qquad (٨\text{-}٣٨)$$

بتطبيق قانون سمبسون للتكامل المركب على المعادلة (٨- ٣٨) نحصل على المعادلة المطولة التالية مع إهمال الجزء الخاص بالخطأ:

$$\begin{aligned}\int_a^b\int_c^d f(x.y)\,dx\,dy \approx \frac{kh}{9}\Bigg\{&\left[f(x_0.y_0)+2\sum_{i=1}^{\left(\frac{n}{2}\right)-1} f(x_{2i}.y_0)+4\sum_{i=1}^{n/2} f(x_{2i-1}.y_0)+f(x_n.y_0)\right]\\&+2\left[\sum_{j=1}^{\left(\frac{m}{2}\right)-1} f(x_0.y_{2j})+2\sum_{j=1}^{\left(\frac{m}{2}\right)-1}\sum_{i=1}^{\left(\frac{n}{2}\right)-1} f(x_{2i}.y_{2j})+4\sum_{j=1}^{\left(\frac{m}{2}\right)-1}\sum_{i=1}^{n/2} f(x_{2i-1}.y_{2i})+\sum_{j=1}^{\left(\frac{m}{2}\right)-1} f(x_n.y_{2j})\right]\\&+4\left[\sum_{j=1}^{m/2} f(x_0.y_{2j-1})+2\sum_{j=1}^{m/2}\sum_{i=1}^{\left(\frac{n}{2}\right)-1} f(x_{2i}.y_{2j-1})+4\sum_{j=1}^{m/2}\sum_{i=1}^{n/2} f(x_{2i-1}.y_{2i-1})+\sum_{j=1}^{m/2} f(x_n.y_{2j-1})\right]\\&+\left[f(x_0.y_m)+2\sum_{i=1}^{\left(\frac{n}{2}\right)-1} f(x_{2i}.y_m)+4\sum_{i=1}^{n/2} f(x_{2i-1}.y_m)+f(x_n.y_m)\right]\Bigg\}\end{aligned} \qquad (٨\text{-}٣٩)$$

يمكن تطبيق نفس الطريقة لإيجاد التكاملات الثلاثية والأعلى منها.

بالطبع سيكون من الصعب تناول أمثلة على التكامل الثنائى وتنفيذها خطوة بخطوة تبعا للمعادلة (٨- ٣٩) ولكن طبعا لحسن الحظ فإن ماتلاب به دوال لإجراء التكاملات الثنائية.

الدالة quad2d(fun,a,b,c,d) تقابل الدالة quad( ) التى سبق دراستها فى حالة المتغير الواحد. بالطبع fun هو إشارة للدالة المطلوب تكاملها و a و b و c و d هى حدود التكامل كما ذكرنا.

مثال ٨-١٠

استخدم ماتلاب فى حساب تكامل الدالة f(x,y)=ysin(x)+xcos(y) فى المدى $\pi \le x \le 2\pi$ و $0 \le y \le \pi$ بإدخال هذه الدالة فى الأمر quad2d( ) فى مساحة عمل الماتلاب نحصل على ما يلى:

```
Q = quad2d(@(x,y) y.*sin(x)+x.*cos(y),pi,2*pi,0,pi)
Q =
  -9.8696
```

مع العلم أن التكامل الحقيقى لهذه الدالة يساوى $-\pi 2$.

كمثال آخر من مساعدة ماتلاب سنحسب تكامل الدالة التالية $f(x) = \frac{1}{\sqrt{(x+y)}\,(1+x+y)^2}$ فى المدى $0 \le x \le 1$ و $0 \le y \le 1-x$ ، مع العلم أن قيمة هذا التكامل هى $(\pi/4)-0.5$.

```
>> fun = @(x,y) 1./( sqrt(x + y) .* (1 + x + y).^2 );
>> ymax = @(x) 1 - x;
 >>Q = quad2d(fun,0,1,0,ymax);
Q =
  0.2854
```

٨-١٤ تمارين

١- أوجد التفاضل الفرقى f'(x) الأمامى والخلفى للنقاط التالية:

x	0.5	0.6	0.7
f(x)	0.4794	0.5646	0.6442

٢- إذا كانت النقاط الموجودة فى جدول التمرين ١ هى عينات للدالة f(x)=sin(x). إحسب الخطأ بين التفاضل الفرقى الناتج فى التمرين ١ والتفاضل التحليلى للدالة f(x)=sin(x).

٣- أعد التمرين ١ للنقاط التالية:

x	0.0	0.2	0.4
f(x)	0.00000	0.74140	1.3718

٤- إذا كانت النقاط الموجودة فى جدول التمرين ٣ هى عينات للدالة $f(x)=e^x-2x^2+3x-1$. إحسب الخطأ بين التفاضل الفرقى الناتج فى التمرين ٣ والتفاضل التحليلى للدالة $f(x)= e^x-2x^2+3x-1$.

٥- استخدم طريقة التفاضل الفرقى المركزى لتحديد التفاضل التقريبى لنقاط الجدول التالى:

x	1.1	1.2	1.3	1.4
f(x)	9.025013	11.02318	13.46374	16.44465

٦- أعد التمرين ٥ لنقاط الجدول التالى:

x	8.1	8.3	8.5	8.7
f(x)	16.94410	17.56492	18.19056	18.82091

٧- البيانات الموجودة فى جدول تمرين ٥ مأخوذة كعينات من الدالة $f(x)=e^{2x}$ إحسب التفاضل الحقيقى لهذه الدالة وقارن بالتفاضل التقريبى فى تمرين ٥.

٨- البيانات الموجودة فى جدول تمرين ٦ مأخوذة كعينات من الدالة $f(x)=x\ln(x)$ إحسب التفاضل الحقيقى لهذه الدالة وقارن بالتفاضل التقريبى فى تمرين ٦.

٩- أوجد التفاضل الثانى $f''(x)$ للدالة $f(x)=\cos(\pi x)$ مستخدما قيم الدالة $f(x)$ عند النقاط $x=0.25$ و $x=0.5$ و $x=0.75$، لحساب التفاضل الثانى عند النقطة $x=0.5$. قارن النتيجة التى حصلت عليها بالتفاضل الثانى الحقيقى للدالة $f(x)$.

١٠- أعد المثال ٨-٣ مستخدما معادلة لاجرانج من الدرجة الثانية $(n=2)$ موضحا أن التفاضل الناتج عند النقطة x_0 سيكون على الصورة: $L'(x) = \frac{-3f(x)+4f(x+h)-f(x+2h)}{2h}$ والذى يعتبر تقريب من الدرجة الثانية فى اتجاه واحد للتفاضل الأول.

١١- أعد التمرين السابق ولكن بإجراء التفاضل عند x_1 بدلا من x_0 حيث سيكون التفاضل الناتج على الصورة: $L'(x) = \frac{f(x+h)-f(x-h)}{2h}$ وهو التفاضل المركزى الأول.

١٢- أوجد تفاضل الدالة $f(x)=\sin(x^2)$ فى المدى $x=0$ حتى $x=\pi$ مستخدما دالة ماتلاب diff(x) وقارن النتيجة التى تحصل عليها مع التفاضل الحقيقى (التحليلى) للدالة f(x) وارسم كل منهما عند $h=0.1$ و $h=0.01$.

١٣- استخدم قانون النقطة المتوسطة لإجراء التكاملات التالية:

(أ) $\int_{0.5}^{1} x^4 dx$

(ب) $\int_{0}^{0.5} \frac{2}{x-4} dx$

(ت) $\int_{1}^{1.5} x^2 \ln(x) dx$

(ث) $\int_{0}^{1} x^2 e^{-x} dx$

(ج) $\int_{0}^{\pi/4} x\sin(x) dx$

١٤- أعد تمرين ١٣ مستخدما قانون شبه المنحرف.

١٥- أعد تمرين ١٣ مستخدما قانون سمبسون.

١٦- استخدم قانون شبه المنحرف لحساب قيم التكاملات الآتية باستخدام نقاط التقسيم الموضحة مع كل منها وقارن النتائج التى تحصل عليها بالقيمة الحقيقية (التحليلية) لهذا التكامل. تحقق من هذه الإجابة أيضا باستخدام دالة ماتلاب trapz(x,y,n).

(أ) $n = 4$. $\int_{0}^{2} x dx$

(ب) $n = 4$. $\int_{0}^{2} x^3 dx$

(ت) $.n = 4$ $\int_0^2 x^4 dx$

(ث) $.n = 4$ $\int_1^2 \frac{1}{x^2} dx$

١٧- استخدم قانون سمبسون لحساب قيم التكاملات الآتية باستخدام نقاط التقسيم الموضحة مع كل منها وقارن النتائج التى تحصل عليها بالقيمة الحقيقية (التحليلية) لهذا التكامل. تحقق من هذه الإجابة أيضا باستخدام دالة ماتلاب .quad('f',a,b)

(أ) $.n = 4$ $\int_0^2 x^2 dx$

(ب) $.n = 4$ $\int_0^\pi \sin(x) dx$

(ت) $.n = 4$ $\int_0^1 \frac{1}{x^2+1} dx$

١٨- إذا كان تطبيق قانون شبه المنحرف على التكامل $\int_0^2 f(x)\, dx$ يعطى القيمة 4، وتطبيق قانون سمبسون 1/3 على نفس التكامل يعطى القيمة 2، فما هى قيمة f(1)؟

١٩- إذا كان تطبيق قانون شبه المنحرف على التكامل $\int_0^2 f(x)\, dx$ يعطى القيمة 5، وتطبيق قانون النقطة المتوسطة على نفس التكامل يعطى القيمة 4، فما هى القيمة التى ستحصل عليها بتطبيق قانون سمبسون 1/3؟

٢٠- الدالة f(x) معطاة فى الصورة المجدولة التالية:

x	1.8	2.0	2.2	2.4	2.6
f(x)	3.12014	4.42569	6.04241	8.03014	10.46675

إحسب التكامل $\int_{1.8}^{2.6} f(x)\, dx$ مستخدما الطرق التالية: ١- النقطة المتوسطة ٢- قانون شبه المنحرف ٣- قانون سمبسون.

٢١- إحسب التكامل $\int_0^2 x^2 e^{-x^2}\, dx$ مستخدما عرض مقطع h=0.5، وباستخدام الطرق التالية:

(أ) طريقة شبه المنحرف المركبة

(ب) طريقة سمبسون المركبة

٢٢- بفرض أن f(0.25)=f(0.75)=α فما هى قيمة α إذا كان قانون شبه المنحرف المركب هو القانون المستخدم فى التكامل $\int_0^1 f(x)\, dx$ يعطى القيمة 2 مع n=2 ويعطى القيمة 1.75 مع n=4.

٢٣- أحد سيارات السباق تدور دورة السباق بالكامل فى ٤٨ ثانية، ولقد تم تسجيل سرعة السيارة كل ٦ ثوان كما فى الجدول التالى من بداية الدورة حتى نهايتها. إحسب مسافة هذه الدورة.

الزمن	0	6	12	18	24	30	36	42	48	54	60	66	72	78	84
السرعة	124	134	148	156	147	133	121	109	99	85	78	89	104	116	123

٢٤- استخدم ماتلاب لحساب التكامل ثنائى الأبعاد لكل من الدوال التالية بفرض n=m=6:

(أ) $\int_{2.1}^{2.5} \int_{1.2}^{1.4} xy^2\, dydx$

(ب) $\int_0^{0.5} \int_0^{0.5} e^{y-x}\, dydx$

(ت) $\int_{2}^{2.2}\int_{x}^{2x}(x^2+y^2)\,dydx$

(ث) $\int_{2}^{1.5}\int_{0}^{x}(x^2+\sqrt{y})\,dydx$

الفصل ٩

حل المعادلات التفاضلية بالطرق العددية

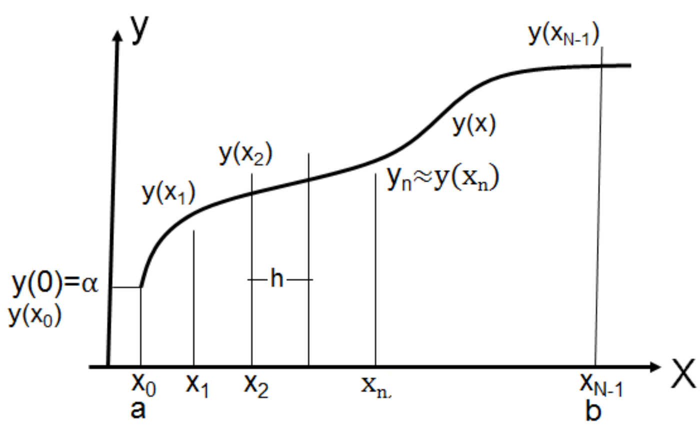
الفصل ٩

حل المعادلات التفاضلية بالطرق العددية

Numerical solution of ordinary differential equations

تستخدم المعادلات التفاضلية لنمذجة المسائل أو المشاكل التى تحتوى على تغير فى أحد المتغيرات التابعة بالنسبة لمتغير آخر مستقل. مثل هذه المشاكل تتطلب الحل لأحد مشاكل القيمة الابتدائية initial value problem، أى الحل لمعادلة تفاضلية تحقق قيم ابتدائية معينة.

فى العديد من مشاكل الحياة العملية، تكون هذه المعادلة التفاضلية التى تمثل نموذج المشكلة معقدة جدا بحيث يصعب إيجاد حل تحليلى صحيح لها، وفى هذه الحالة يتم اتباع واحدة من طريقتين لإيجاد حل تقريبى لهذه المشكلة: الطريقة الأولى هى محاولة تبسيط المعادلة التفاضلية إلى صورة يمكن حلها تحليليا واستخدام هذا الحل الصحيح للمعادلة المبسطة ليمثل حل المشكلة الأصلية. الطريقة الثانية وهى التى سنتبعها فى هذا الفصل ستحتوى على طرق عديدة يمكن بها إيجاد تقريب عددى مباشر لحل المعادلة الأصلية، وهذه الطريقة بالطبع تعطى دقة أفضل فى التقريب من الطريقة الأولى. هذه الطرق لن تعطى حلا تقريبيا متصلا أو مستمرا للمعادلة التفاضلية ولكن هذا الحل سيكون عند نقط محددة ومتباعدة بمسافات تكون فى الغالب متساوية.



شكل ٩–١ الحل العددى التقريبى للمعادلة التفاضلية $y'=f(x,y)$ عند النقاط المتساوية التباعد x_0 حتى x_{N-1} فى المدى [a b].

الصورة العامة للمعادلات التى سنتعرض لحلها فى هذا الفصل هى:

$$\frac{dy}{dx} = y'(x) = f(x.y(x)) \quad \text{حيث} \quad a\leq x\leq b \qquad (٩-١)$$

مع القيمة الابتدائية التالية: $y(a)=\alpha$. هنا الدالة f() تمثل علاقة حسابية بين المتغير المستقل x والمتغير التابع y(x)، والمقصود من حل المعادلة التفاضلية (٩– ١) هو إيجاد y(x) التى تحقق هذه المعادلة والتى يجب أن تكون مستمرة فى المدى [a b]. المتغير المستقل x (الكثير من المراجع تستخدم المتغير t وهو الزمن على أنه المتغير المستقل، ولكن ذلك لن يغير من الأمر شيء) يجب أن يكون له مدى محدد وهو [a b] حيث a تمثل بداية تغير y(x) مع x، وقيمة y(x) عند نقطة البداية تلك تسمى القيمة الابتدائية $y(a)=\alpha$. الحد الثانى لمدى تغير x وهو b يجب ألا يمتد إلى المالانهاية. هذا الشكل للمعادلة التفاضلية يسمى المعادلة التفاضلية العادية ordinary differential equation, ODE، وهى من الدرجة الأولى لأن رتبة أعلى تفاضل فيها هى التفاضل الأول، ويفترض أن تكون خطية فى y. أحيانا تسمى هذه المعادلة بمسألة كوشى Cauchy أو مسألة القيمة الابتدائية كما ذكرنا.

الحل العددى أو التقريبى للمعادلة (٩ –١) سيتمثل فى إيجاد تتابع من القيم للمتغير المستقل x_0 و x_1 و ... وتتابع من القيم للمتغير التابع y_0 و y_1 و ... بحيث تمثل القيمة y_n حلا للمعادلة عند x_n على الصورة التالية:

$$y_n \approx y(x_n) \quad \text{لكل القيم } n=0, 1, \dots$$

شكل ٩–١ يبين هذا الحل العددى التقريبي حيث يوجد هناك خطأ معين عند كل نقطة x_n يتوقف على الطريقة المستخدمة. للحصول على معادلة مستمرة للقيم y_n يمكن استخدام أحد طرق الاستيفاء interpolation.

أنظر إلى المعادلة التالية:

$$y'(x) = -\alpha(y - \sin(x)) + \cos(x) \quad \text{حيث } x>0 \text{ و } y(0)=1$$

هذه المعادلة تعتبر معادلة تفاضلية خطية من الدرجة الأولى، وهى خطية لوجود y فقط وليس أحد قواها أو مشتقاتها. بينما المعادلة التالية:

$$y' = y^2 \quad \text{حيث } x>0 \text{ و } y(0)=1.$$

هذه المعادلة تعتبر معادلة تفاضلية غير خطية من الدرجة الأولى لوجود العنصر y^2. فى العادة سنكتب y بدلا من y(x) و y' بدلا من $y'(x)$ بغرض التبسيط فقط.

بعد ذلك سنتعرض لحل نظام من المعادلات التفاضلية من الدرجة الأولى على الصورة التالية:

$$\begin{aligned} \frac{dy_1}{dx} &= f_1(x.y_1.y_2 \dots .y_n) \\ \frac{dy_2}{dx} &= f_2(x.y_1.y_2 \dots .y_n) \\ &- - - - - - - - - - - - - - \\ \frac{dy_n}{dx} &= f_n(x.y_1.y_2 \dots .y_n) \end{aligned} \qquad (٩–٢)$$

حيث أيضا $a \le x \le b$ مع القيم الابتدائية التالية: $y_1(a)=\alpha_1$ و $y_2(a)=\alpha_2$ و ... و $y_n(a)=\alpha_n$.

سنقدم أيضا علاقة نظام من المعادلات التفاضلية من الدرجة الأولى مثل النظام السابق مع المعادلة التفاضلية من الدرجة n التى يمكن كتابتها على الصورة التالية:

$$y^{(n)} = f(x.y.y'.y'' \dots .y^{(n-1)}) \qquad (٩-٣)$$

حيث y(n) هو التفاضل من الدرجة n للمتغير y بالنسبة للمتغير x، و a≤x≤b مع القيم الابتدائية التالية:

$$y(a) = \alpha_0.\ y'(a) = \alpha_1. \dots\ y^{(n-1)}(a) = \alpha_{n-1}.$$

٩-١ طريقة مفكوك تايلور لحل المعادلات التفاضلية من الدرجة الأولى

كما رأينا مسبقا فإن مفكوك تايلور كان هو الأساس للكثير من طرق الحل العددى، والحل العددى للمعادلات التفاضلية ليس استثناء من ذلك حيث سيكون مفكوك تايلور هو الأساس للحل العددى للمعادلات التفاضلية أيضا وهو الأساس لطريقة أويلر المستخدمة بكثرة فى هذا المجال كما سنرى. ينص مفكوك تايلور على أنه يمكن كتابة الدالة y(x) عند نقطة (x+h) تبعد عن x بمقدار h بالمعادلة التالية:

$$y(x+h) = y(x) + y'(x)h + \frac{1}{2!}y''(x)h^2 + \frac{1}{3!}y'''(x)h^3 + \cdots . + \frac{1}{n!}y^{(n)}(x)h^n \qquad (٩-٤)$$

حيث y' و y'' و y''' و y(n) هى التفاضل من الدرجة الأولى والثانية والثالثة وحتى الدرجة n، و h هى عرض المقطع حيث تم تقطيع المدى [a b] إلى عدد N من المقاطع بحيث h=(b-a)/N، كما فى شكل ٩-١.

فى العادة يتم الاكتفاء بعدد محدد من المقادير فى الطرف الأيمن للمعادلة (٩ -٤) حيث سيتوقف على ذلك دقة الطريقة المستخدمة. طريقة أويلر تكتفى بمقدارين فقط من هذه المعادلة والباقى يعتبر خطأ حيث فى هذه الحالة سيكون الخطأ متناسبا مع h^2، وكما ذكرنا من قبل فإننا نقول أن الخطأ من الدرجة الثانية $O(h^2)$. وعلى ذلك فإن المعادلة (٩ -٤) تؤول إلى:

$$y(x+h) = y(x) + y'(x)h \qquad (٩-٥)$$

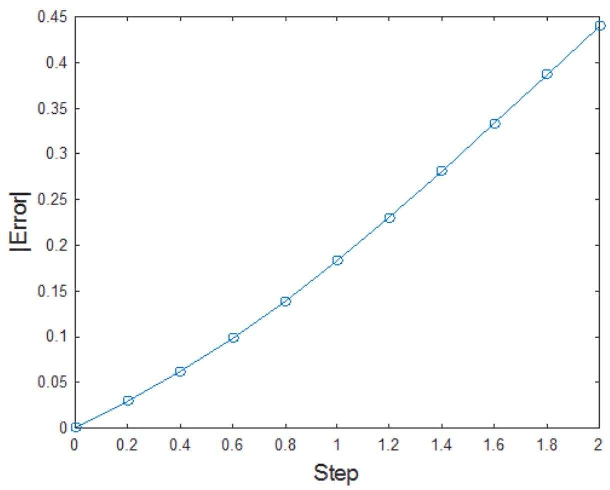
بالتعويض عن $y'(x)$ من المعادلة (٧ -١) فى المعادلة (٧ -٥) نحصل على:

$$y(x+h) = y(x) + f(x.y)h \qquad (٩-٦)$$

بذلك أمكن الحصول على الدالة y(x) عند نقطة تبعد h عن النقطة x بدلالة قيمة الدالة عند النقطة x وهى y(x) وقيمة الدالة f(x,y) وعرض مقطع التقسيم h. يمكن تعميم المعادلة (٩- ٦) لتصبح على الصورة:

$$y(x_{n+1}) = y(x_n) + f(x.y)h$$

ويمكن كتابة هذه المعادلة فى صورة أكثر عمومية كالتالى:



شكل ٩-٢ القيمة المطلقة للخطأ مع تزايد خطوات حل المعادلة التفاضلية

$$y_{n+1} = y_n + f(x.y_n)h \qquad (٩-٧)$$

حيث n=0,1,1,...,N

يبدأ الحل العددى لهذه المعادلة بالتعويض عن y_0 فى المعادلة (٩- ٧) وبمعلومية f( ) و h يمكن حساب y_1. بمعلومية y_1 يمكن حساب y_2 وهكذا بمعلومية y_n يمكن الحصول على y_{n+1}. هذا النوع من طرق الحل يسمى طرق الحل ذات الخطوة الواحدة one step حيث أنه يوجد خطوة واحدة بين قيمة الحل عند x_n و x_{n+1}. تسمى هذه الطريقة أيضا بالطريقة الصريحة explicit حيث أن الطرف الأيسر من المعادلة (٩- ٧) يحتوى فقط على y_{n+1} على العكس من بعض الطرق الأخرى التى يوجد فيها العنصر y_{n+1} صريحا فى الطرف الأيسر وضمنيا فى الطرف الأيمن ($y_{n+1} = y_n + f(x.y_{n+1})h$) وفى هذه الحالة تكون المعادلة التفاضلية غير خطية وتسمى بالطريقة ذات الخطوة الواحدة الضمنية explicit one step method، حيث يكون الحل فى هذه الحالة أكثر تكلفة حسابيا. لذلك إذا لم تكن مضطرا لاستخدام الطرق الضمنية فإنه من المفضل استخدام الطرق الصريحة.

مثال ٩-١

استخدم طريقة أويلر للحصول على حل تقريبى للمعادلة التفاضلية التالية:

$$y' = y - x^2 + 1$$

حيث $0 \le x \le 2$ والقيمة الابتدائية y(0)=0.5.

بفرض N=10 فإن h=(2-0)/10=0.2 وبالتالى يمكن كتابة الحل التقريبى من المعادلة (٩- ٧) كما يلى مع العلم بأن الدالة f(x,y)=y-x2+1:

$$y(x_{n+1}) = y(x_n) + (y(x_n) - x_n^2 + 1)h$$

حيث n تتغير من 0 حتى 9 و $x_0=0$ و $y(x_0)=0.5$.

الحل التحليلى الصحيح لهذه المعادلة التفاضلية هو:

$$y = (x+1)^2 - 0.5e^x$$

بكتابة برنامج بسيط يمكن كتابة قيم الحل العددى التقريبى والحل التحليلى عند هذه القيم العشرة للمتغير x فى المدى [0 2] كما يلى:

step	true	approx.	|Error|
0	0.5000	0.5000	0
0.2000	0.8000	0.8293	0.0293
0.4000	1.1520	1.2141	0.0621
0.6000	1.5504	1.6489	0.0985
0.8000	1.9885	2.1272	0.1387
1.0000	2.4582	2.6409	0.1827
1.2000	2.9498	3.1799	0.2301
1.4000	3.4518	3.7324	0.2806
1.6000	3.9501	4.2835	0.3334
1.8000	4.4282	4.8152	0.3870
2.0000	4.8658	5.3055	0.4397

لاحظ من الجدول السابق ومن شكل ٩-٢ أن القيمة المطلقة للخطأ تراكمية، بمعنى أنها تبدأ من الصفر عند القيمة الابتدائية للمعادلة التفاضلية وتزداد مع زيادة الخطوات أو التقريب حتى يصل الخطأ إلى أكبر قيمة عند الطرف الثانى لمدى التقريب.

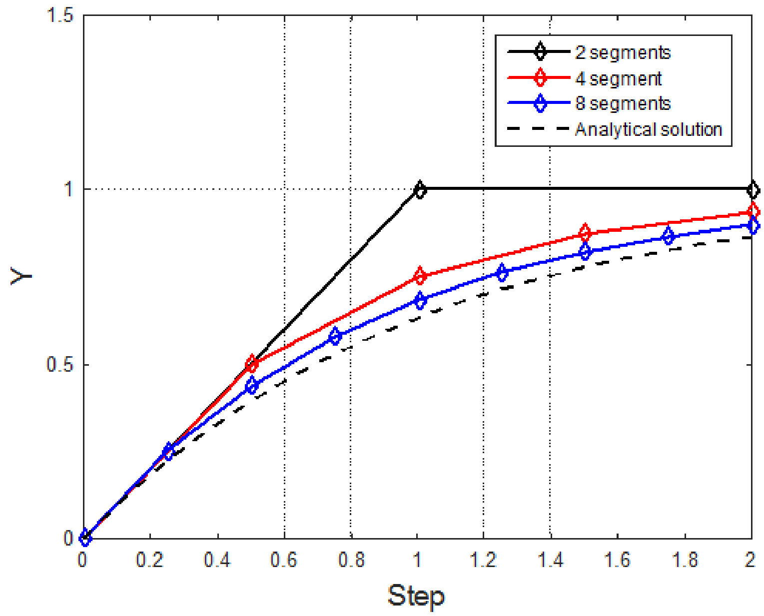
فى هذا المثال تم تقطيع المدى [a b] إلى عدد معين من المقاطع (١٠ فى هذا المثال) وبالطبع مع زيادة عدد المقاطع نتوقع أن يقترب الحل العددى من الحل التحليلى كما يوضح المثال التالى.

مثال ٩-٢

استخدم طريقة أويلر للحصول على حل تقريبى للمعادلة التفاضلية التالية:

$$y'(x) = 1 - y(x)$$

حيث $0 \le x \le 2$ والقيمة الابتدائية y(0)=0.0 بفرض N=2 مرة و N=4 مرة و N=8 مرة أخرى، وقارن ذلك بالحل التحليلى الذى هو y(x)=1-e-x. شكل ٩-٣ يبين الحل العددى فى الحالات الثلاث المطلوبة ويبين أيضا الحل التحليلى حيث نلاحظ أن منحنى الثمانى مقاطع هو الأقرب للحل التحليلى.



شكل ٩–٣ تأثير زيادة المقاطع فى المدى [a b] على دقة الحل العددى

البرنامج التالى مباشر وبسيط يبين كيفية الحصول على شكل ٩–٣.

```
% program to approximate the solution of an ordinary diff. equation
% the ODE is y'(x)=1-y(x), showing effect of increasing number of segments
clear;
N1=2;N2=4;N3=8;
h1=2/N1;h2=2/N2;h3=2/N3;
y1(1)=0;y2(1)=0;y3(1)=0;
for i=1:N1
    y1(i+1)=(1-h1)*y1(i)+h1;
end
for i=1:N2
    y2(i+1)=(1-h2)*y2(i)+h2;
end
for i=1:N3
    y3(i+1)=(1-h3)*y3(i)+h3;
end
 i=0:h1:2;
plot(i,y1,'kd-', 'linewidth',1.5); hold on;
i=0:h2:2;
plot(i,y2,'rd-', 'linewidth',1.5); hold on;
i=0:h3:2;
plot(i,y3,'bd-', 'linewidth',1.5);hold on;
i=linspace(0,2);
y4=1-exp(-i);
plot(i,y4,'k--', 'linewidth',1.5);grid
ylabel('Y', 'fontsize',14)
xlabel('Step', 'fontsize',14)
axis([0 2 0 1.5]);
legend('2 segments','4 segment', '8 segments', 'Analytical solution')
```

٩-٢ طريقة شبه المنحرف لحل المعادلات التفاضلية

طريقة أخرى لحل المعادلات التفاضلية التى على الصورة $y'(x) = f(x.y)$ حيث x فى المدى [a b] والقيمة الابتدائية هى y0=α يمكن أن تأتى من تكامل طرفى المعادلة التفاضلية كما يلى:

$y'(x) = f(x.y)$ بتكامل الطرفين:

$$\int_{x_k}^{x_{k+1}} y'(x)dx = \int_{x_k}^{x_{k+1}} f(x.y)\,dx$$

$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} f(x.y)\,dx$$

وبالتالى يمكن كتابة:

(٩-٨) $$y(x_{k+1}) = y(x_k) + \int_{x_k}^{x_{k+1}} f(x.y)\,dx$$ حيث y0=α .

إذا افترضنا أن الدالة f(x,y) كانت ثابتة فى المدى $[x_k\ x_{k+1}]$ وهو المدى [a b] الذى يساوى h، فإن التكامل فى المعادلة (٩- ٨) سيساوى hf(x,y) وسيكون حل المعادلة التفاضلية فى هذه الحالة يساوى:

(٩-٩) $$y(x_{k+1}) = y(x_k) + hf(x.y)$$

وهو نفس الحل الذى حصلنا عليه بطريقة أويلر (أو تايلور) فى المعادلة (٩- ٧).

التكامل الذى فى المعادلة (٩- ٨) يمكن حله بطريقة شبه المنحرف التى درسناها فى معرض الحديث عن الطرق المختلفة للتكامل العددى، وفى هذه الحالة يمكن كتابة المعادلة (٩- ٨) كالتالى وبعد تطبيق طريقة شبه المنحرف:

(٩-١٠) $$y_{k+1} = y_k + \frac{h}{2}[f(x_k.y_k) + f(x_{k+1}.y_{k+1})]$$

لاحظ أن الجانب الأيمن من المعادلة (٩- ١٠) يحتوى على y_{k+1} داخل الدالة f( ) بحيث يصعب نقل هذا المقدار إلى الجانب الأيسر حتى يمكن حساب الحل، ولذلك فإن المعادلة (٩- ١٠) أصبحت غير خطية ولا يمكن حلها فى أغلب الأحوال. وكما ذكرنا من قبل فإن طريقة شبه المنحرف تعتبر مثالا على الطرق أحادية الخطوة الضمنية نتيجة وجود y_{k+1} ضمنيا فى الطرف الأيمن للحل فى المعادلة (٩- ١٠).

أحد الطرق للخروج من هذا التعقيد هو بوضع $y_{k+1} \approx y_k + hf(x_k.y_k)$ فى الطرف الأيمن من المعادلة (٩- ١٠) وبالتالى يمكن كتابة هذه المعادلة كما يلى:

(٩-١١) $$y_{k+1} = y_k + \frac{h}{2}[f(x_k.y_k) + f(x_{k+1}.y_k + hf(x_k.y_k))]$$

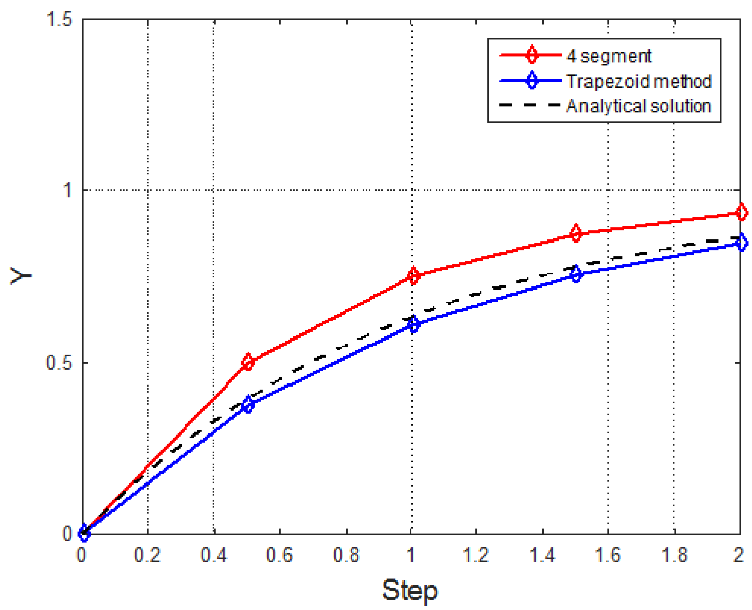
هذه الطريقة تسمى بطريقة هيون Heun لحل المعادلة التفاضلية الخطية من الدرجة الأولى. خطأ القطع فى هذه الطريقة يتناسب مع h^2، أى أنه $O(h^2)$ كما رأينا فى الفصل ٦ مع طريقة شبه المنحرف للتكامل، وهذا بالطبع أفضل من طريقة أويلر التى كان يتناسب الخطأ فيها مع h، أى (O(h)).

مثال ٩–٣

لتوضيح ميزة هذه الطريقة سنقوم بحل المثال ٩– ٢ وهو حل المعادلة التفاضلية $y'(x) = 1 - y(x)$ فى المدى [0 2] وحيث $y_0=0$. فى هذه الحالة $f(x,y)=1-y(x)$. وبالتعويض عن ذلك فى المعادلة (٩– ١١) وبعد القليل من الاختصارات يمكن كتابة حل هذه المعادلة التفاضلية بطريقة شبه المنحرف على الصورة التالية:

$$y_{k+1} = \left(1 - h + \frac{h^2}{2}\right) y_k + h - \frac{h^2}{2} \tag{٩–١٢}$$

البرنامج التالى يوضح هذا الحل مع مقارنة الناتج بطريقة أويلر ذات الأربع نقاط وبالحل التحليلى كما فى شكل ٩–٤.



شكل ٩–٤ مقارنة الحل بطريقة أويلر ذات الأربع نقاط وطريقة شبه المنحرف مع الحل التحليلى. لاحظ اقتراب طريقة شبه المنحرف من الحل الصحيح

```
% program to approximate the solution of an ordinary diff. equation
% the ODE is y'(x)=1-y(x). Using trapezoid method
clear;
N2=4; h2=2/N2;
y2(1)=0;y3(1)=0;
for i=1:N2 % solution with Euler 4 points
    y2(i+1)=(1-h2)*y2(i)+h2;
end
i=0:h2:2;
plot(i,y2,'rd-', 'linewidth',1.5); hold on;
for i=1:N2   %solution using trapezoidal method
    y3(i+1)=y3(i)*(1-h2+h2^2/2)+h2-h2^2/2;
end
i=0:h2:2; % plotting result of trapezoidal method
plot(i,y3,'bd-', 'linewidth',1.5); hold on;
```

```
i=linspace(0,2);  % plotting real solution
y4=1-exp(-i);
plot(i,y4,'k--', 'linewidth',1.5);grid
ylabel('Y', 'fontsize',14)
xlabel('Step', 'fontsize',14)
axis([0 2 0 1.5]);
legend('4 segment', 'Trapezoid method' ,'Analytical solution')
```

لاحظ من شكل ٩-٤ كيف أن الحل باستخدام طريقة شبه المنحرف أكثر قربا من الحل الصحيح من طريقة أويلر ذات الأربع نقاط. لاحظ أنه تم استخدام نفس العدد من النقاط (٤ نقاط) فى الطريقتين.

٩-٣ طريقة النقطة المتوسطة

بنفس الطريقة كما فعلنا مع طريقة شبه المنحرف يمكن حل المعادلة التفاضلية التى على الصورة $y'(x) = f(x.y)$ حيث x فى المدى [a b] والقيمة الابتدائية هى $y_0=\alpha$ عن طريق تكامل طرفى المعادلة التفاضلية قبل وبعد النقطة x كما يلى:

$$y'(x) = f(x.y) \quad \text{بتكامل الطرفين:}$$
$$\int_{x_{k-1}}^{x_{k+1}} y'(x)dx = \int_{x_{k-1}}^{x_{k+1}} f(x.y)\,dx$$
$$y(x_{k+1}) - y(x_{k-1}) = \int_{x_{k-1}}^{x_{k+1}} f(x.y)\,dx$$

وبالتالى يمكن كتابة:

$$y(x_{k+1}) = y(x_{k-1}) + \int_{x_{k-1}}^{x_{k+1}} f(x.y)\,dx \quad \text{حيث } y0=\alpha \;. \qquad (\text{٩-١٣})$$

بتقريب الدالة f(x,y) فى المعادلة (٩- ١٣) بخط أفقى يمر بالنقطة المتوسطة x من النقطة x-1 حتى النقطة x+1 فإن التكامل فى هذه المعادلة يمكن كتابته كمساحة للمستطيل الذى عرضه 2h وارتفاعه الدالة f(x,y)، وبالتالى يمكن كتابة المعادلة (٩- ١٣) كما يلى:

$$y_{k+1} = y_{k-1} + 2hf(x.y) \qquad (\text{٩-١٤})$$

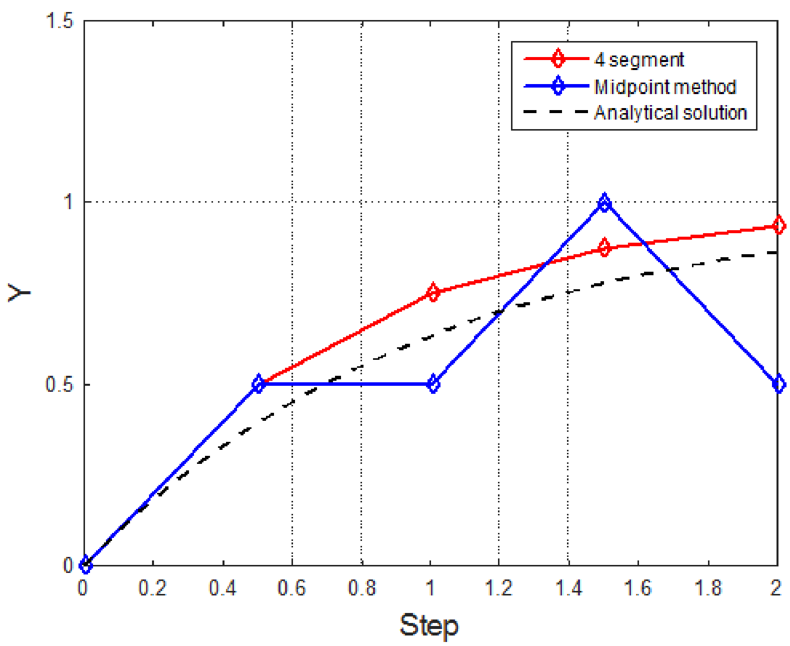
المعادلة (٩- ١٤) توضح أن طريقة النقطة المتوسطة تعتبر طريقة ثنائية الخطوة حيث أن y_{k+1} تحتاج لحسابها إلى y_k و y_{k-1} وهى صريحة explicit نتيجة وجود y_{k+1} صراحة فى الطرف الأيسر فقط.

مثال ٩-٤

بتطبيق طريقة النقطة المتوسطة على المعادلة التفاضلية فى مثال ٩-٣ حيث $y'(x) = 1 - y(x)$ والقيمة الابتدائية $y_0=0$ ، وحيث أننا سنحتاج نقطتان فى البداية فإننا سنحسب النقطة التالية للقيمة الابتدائية باستخدام طريقة أويلر،

ثم نستخدام هذه القيمة والقيمة الابتدائية للتعويض فى المعادلة (٩– ١٤) لحساب القيم الجديدة للدالة f(x,y) بطريقة النقطة المتوسطة. نتيجة لذلك فإن حل المعادلة بهذه الطريقة سيكون أسوأ من طريقة أويلر كما يبين ذلك البرنامج التالى وشكل ٩–٥.

```
% program to approximate the solution of an ordinary diff. equation
% the ODE is y'(x)=1-y(x). Using midpoint method
clear;
N2=4; h2=2/N2;
y2(1)=0;y3(1)=0;y3(2)=h2;
for i=1:N2 % solution with Euler 4 points
    y2(i+1)=(1-h2)*y2(i)+h2;
end
i=0:h2:2;
plot(i,y2,'rd-', 'linewidth',1.5); hold on;
for i=2:N2   %solution using midpoint method
    y3(i+1)=y3(i-1)+2*h2*(1-y3(i));
end
i=0:h2:2; % plotting result of midpoint method
plot(i,y3,'bd-', 'linewidth',1.5); hold on;
i=linspace(0,2);  % plotting real solution
y4=1-exp(-i);
plot(i,y4,'k--', 'linewidth',1.5);grid
ylabel('Y', 'fontsize',14)
xlabel('Step', 'fontsize',14)
axis([0 2 0 1.5]);
legend('4 segment', 'Midpoint method' ,'Analytical solution')
```



شكل ٩–٥ مقارنة بين حل المعادلة التفاضلية $y'(x)=1-y(x)$ باستخدام طريقة أويلر ذات الأربع نقاط وطريقة النقطة المتوسطة مع المقارنة مع الحل التحليلى للمعادلة التفاضلية

٩-٤ طريقة رونج كوتا Runge Kutta

لقد رأينا فى الجزء ٩-٣ كيفية استخدام التكامل فى الحصول على حل للمعادلة التفاضلية $y'(x) = f(x.y(x))$ كما فى المعادلة (٩- ٨) التى سنعيد كتابتها هنا مرة ثانية للتذكرة كالتالى:

$$y(x_{k+1}) = y(x_k) + \int_{x_k}^{x_{k+1}} f(x.y(x))\,dx \qquad (٩-١٥)$$

حيث $y0=\alpha$.

ورأينا كيف أن التعويض عن التكامل فى الطرف الأيمن من المعادلة (٩- ١٥) بتكامل شبه المنحرف من الفصل ٦ قد أعطى ما يسمى بطريقة شبه المنحرف أو طريقة هيون التى كانت أكثر دقة من طريقة أويلر. هنا فى طريقة رونج كوتا يتم اعتماد طريقة سمبسون للتكامل للتعويض عن التكامل الذى فى الطرف الأيمن من المعادلة (٩- ١٥) كما يلى:

$$y(x_{k+1}) = y(x_k) + \frac{h}{6}\left[f(x_k.y(x_k)) + 4f\left(x_{k+1/2}.y(x_{k+1/2})\right) + f(x_{k+1}.y(x_{k+1}))\right] \qquad (٩-١٦)$$

لاحظ وجود $y(x_{k+1})$ و $y(x_{k+1/2})$ فى الطرف الأيمن مما سيحتاج لإجراء تقريبا معينا لجعل المعادلة (٩- ١٦) فى صورة خطية لا يحتوى فيها الطرف الأيمن على الكميات $y(x_{k+1})$ أو $y(x_{k+1/2})$. هذه التقريبات لن نخوض فى تفاصيلها هنا ولكننا سنكتب آخر ما وصلت إليه طريقة رونج كوتا وهو وضع حل المعادلة التفاضلية على الصورة المتسلسلة التالية:

$$y_{k+1} = y_k + \frac{h}{6}(f_{k1} + 2f_{k2} + 2f_{k3} + f_{k4}) \qquad (٩-١٧)$$

حيث:

$$f_{k1} = f(x_k.y_k) \qquad (٩-١٨)$$

$$f_{k2} = f(x_k + \frac{h}{2}.y_k + \frac{h}{2}f_{k1}) \qquad (٩-١٩)$$

$$f_{k3} = f(x_k + \frac{h}{2}.y_k + \frac{h}{2}f_{k2}) \qquad (٩-٢٠)$$

$$f_{k4} = f(x_k + h.y_k + hf_{k3}) \qquad (٩-٢١)$$

هذه الطريقة تسمى طريقة رونج كوتا من الدرجة الرابعة نتيجة استخدام الأربع تقريبات كما فى المعادلة (٩- ١٧). بالمثل كان هناك ما يسمى بطريقة رونج كوتا من الدرجة الثانية والحل بهذه الصورة يمكن كتابته بالطريقة التالية:

$$y_{k+1} = y_k + \frac{1}{2}(f_{k1} + f_{k2}) \qquad (٩-٢٢)$$

حيث:

$$f_{k1} = hf(x_k.y_k)$$
$$f_{k2} = hf(x_{k+1}.y_{k+1})$$

لاحظ أنه بالتعويض عن f_{k1} و f_{k2} فى المعادلة (٩- ٢٢) فإننا نحصل على الحل بطريقة شبه المنحرف التى سبق تقديمها فى المعادلة (٩- ١٠).

بنفس الطريقة أيضا يمكن كتابة الحل بطريقة رونج كوتا من الدرجة الثالثة كما يلى:

$$y_{k+1} = y_k + \frac{h}{6}(f_{k1} + 4f_{k2} + f_{k3}) \qquad (٩-٢٣)$$

حيث:

$$f_{k1} = f(x_k . y_k)$$
$$f_{k2} = f(x_k + \frac{h}{2} . y_k + \frac{h}{2} f_{k1})$$
$$f_{k3} = f(x_k + h . y_k + h f_{k2} - f_{k1})$$

طريقة رونج كوتا هى الأكثر استخداما والأكثر دقة فى حل المعادلات التفاضلية وهى المستخدمة من قبل ماتلاب فى صورة دوال سابقة البرمجة يتم النداء عليها لحل أى معادلة تفاضلية وذلك بتعديلات يسيطة (لن نخوض فى تفاصيله) عليها لزيادة دقتها كما سنرى فى الجزء الخاص بذلك.

مثال ٩-٥

بتطبيق طريقة رونج كوتا على المعادلة التفاضلية فى مثال ٩-٣ حيث $y'(x) = 1 - y(x)$ والقيمة الابتدائية $y_0=0$ ، فإنه تبعا للمعادلات (٩- ١٨) حتى (٩- ٢١) يمكننا كتابة الدوال التالية:

$$f_{k1} = 1 - y(x)$$
$$f_{k2} = 1 - y(x) - \frac{h}{2} fk1$$
$$f_{k3} = 1 - y(x) - \frac{h}{2} fk2$$
$$f_{k4} = 1 - y(x) - h fk1$$

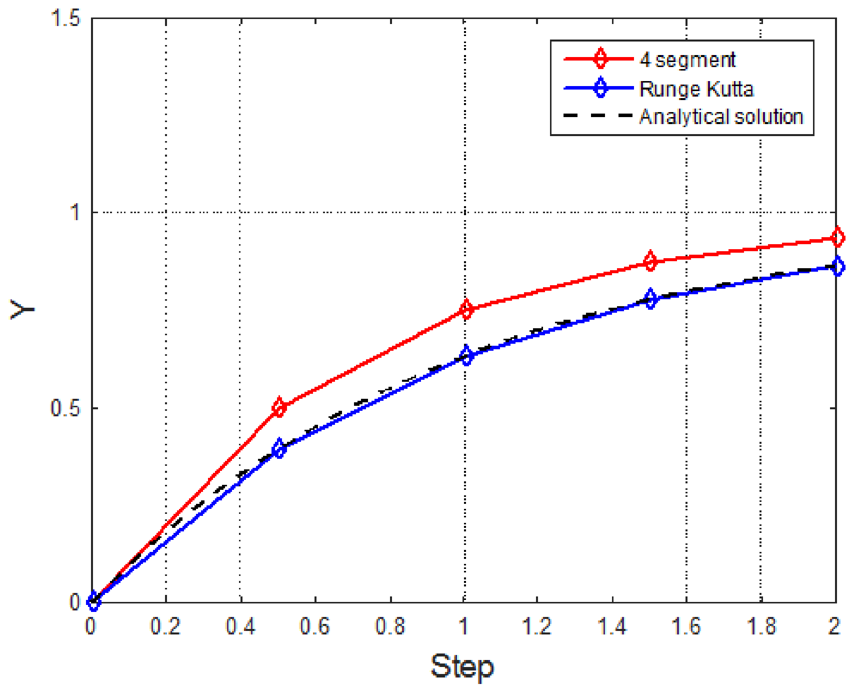
وبالتالى فإن :

$$y_{k+1} = y_k + \frac{h}{6}(f_{k1} + 2f_{k2} + 2f_{k3} + f_{k4})$$

البرنامج التالى وشكل ٩-٦ يبينان جودة طريقة رونج كوتا فى حل المعادلات التفاضلية من الدرجة الأولى وكيف أن النتيجة تكاد تنطبق على الحل التحليلى تقريبا فى هذا المثال.

```
% program to approximate the solution of an ordinary diff. equation
% the ODE is y'(x)=1-y(x). Using Runge Kutta method
clear;
N2=4; h2=2/N2;
y2(1)=0;y3(1)=0;
for i=1:N2 % solution with Euler 4 points
    y2(i+1)=(1-h2)*y2(i)+h2;
end
i=0:h2:2;
plot(i,y2,'rd-', 'linewidth',1.5); hold on;
for i=1:N2   %solution using Runge Kutta method
    fk1=1-y3(i);
    fk2=1-y3(i)-fk1*h2/2;
    fk3=1-y3(i)-fk2*h2/2;
    fk4=1-y3(i)-fk3*h2;
    y3(i+1)=y3(i)+(h2/6)*(fk1+2*fk2+2*fk3+fk4);
end
```

```
i=0:h2:2; % plotting result of Runge Kutta method
plot(i,y3,'bd-', 'linewidth',1.5); hold on;
i=linspace(0,2);  % plotting real solution
y4=1-exp(-i);
plot(i,y4,'k--', 'linewidth',1.5);grid
ylabel('Y', 'fontsize',14)
xlabel('Step', 'fontsize',14)
axis([0 2 0 1.5]);
legend('4 segment', 'Runge Kutta' ,'Analytical solution')
```



شكل ٩-٦ حل المعادلة التفاضلية $y'(x)=1-y(x)$ بطريقة رونج كوتا مع المقارنة بطريقة أويلر والحل التحليلى

٩-٥ المعادلات التفاضلية من الدرجات الأعلى

المعادلات التفاضلية التى تم تقديمها هنا كانت كلها من الدرجة الأولى، والطرق التى تم تقديمها كلها من المفترض أن يتم تطبيقها على المعادلات من الدرجة الأولى. سنوضح فى هذا الجزء كيفية تحويل المعادلات التفاضلية ذات الدرجات الأعلى إلى مجموعة من المعادلات التفاضلية من الدرجة الأولى التى يمكن حلها بأى واحدة من الطرق السابقة.

افترض المعادلة التفاضلية التالية من الدرجة الثانية:

$$y''(x.y) = f(x.y.y') \qquad (٩-٢٤)$$

حيث x>a و $y(a)=y_0$ و $y'(a)=y_0^1$ تمثل القيم الابتدائية. هذه المعادلة تعتبر معادلة تفاضلية من الدرجة الثانية حيث أعلى رتبة للتفاضل هى التفاضل الثانى كما أن الدالة f( ) تكون دالة فى x و y و y' ، (دالة فى كل التفاضلات الأقل رتبة من رتبة تفاضل المعادلة). هذه المعادلة يجب أن يكون لها اثنان من القيم الابتدائية (عدد القيم الابتدائية يساوى رتبة تفاضل هذه المعادلة). الحل لهذه المعادلة هو الحصول على دالة y(x) تحقق المعادلة (٩– ٢٤)، التى نعرف قيمتها وقيمة تفاضلها عند نقطة البداية x=a، وهما القيمتان الابتدائيتان $y(a)=y_0$ و $y'(a)=y_0^1$.

لكى نضع المعادلة التفاضلية التى من الدرجة الثانية فى صورة معادلتين تفاضليتين من الدرجة الأولى سنقوم بإجراء تغيير على أسماء المتغيرات عن طريق إعطاء المتغيرات y(x) و $y'(x)$ (حتى الرتبة التفاضلية الأقل درجة من رتبة المعادلة التفاضلية) أسماء جديدة كما يلى:

$$z_1(x)=y(x) \text{ و } z_2(x)=y'(x)$$

بتفاضل z_1 نحصل على:

$$z_1'(x)=y'(x)=z_2(x)$$

وبالتالى فإن المعادلة التفاضلية الأولى هى $z_1'(x)=z_2(x)$ وهى معادلة تفاضلية خطية من الدرجة الأولى القيمة الابتدائية لها هى $z_1(a)=y(a)=y_0$.

بتفاضل z_2 نحصل على:

$$z_2'(x)=y''(x)=f(x.z_1.z_2)$$

وبالتالى فإن المعادلة التفاضلية الثانية ستكون $z_2'(x)=f(x.z_1.z_2)$، وهى أيضا معادلة تفاضلية خطية من الدرجة الأولى القيمة الابتدائية لها هى $z_2(a)=y'(a)=y_0^1$. وبالتالى أصبح لدينا المعادلتين التفاضليتين التاليتين:

$$z_1'(x)=z_2(x) \quad \text{حيث } z1(a)=y0 \qquad (٩-٢٥)$$

$$z_2'(x)=f(x.y.y') \quad \text{حيث } z_2'(a)=y_0^1 \qquad (٩-٢٦)$$

المعادلتان (٩– ٢٥) و (٩– ٢٦) يسميان نظام من المعادلات التفاضلية من الدرجة الأولى مكون من معادلتين.

افترض المثال التالى المكون من نظام من معادلتين تفاضليتين كل منهما من الدرجة الثانية على الصورة التالية:

$$y''=f(x.y.y'.z.z') \qquad (٩-٢٧)$$

$$z''=g(x.y.y'.z.z') \qquad (٩-٢٨)$$

لاحظ أن كل من y'' و z'' يعتمدان على المتغير المستقل x والمتغيرات y و y' و z و z'. المدى المطلوب البحث عن الحل فيه هو x>a ، كما أن كل معادلة سيكون لها قيمتان ابتدائيتان هما $y(a)=y_0$ و $y'(a)=y_0^1$ للمعادلة الأولى، و $z(a)=z_0$ و $z'(a)=z_0^1$ للمعادلة الثانية. سنبحث هنا عن دالتين y(x) و z(x) تحققان المعادلتين

التفاضلين (٩- ٢٧) و (٩- ٢٨) فى المدى x>a والقيم الابتدائية السابقة. مطلوب تحويل هذا النظام للمعادلتين التفاضليتين من الدرجة الثانية إلى نظام من المعادلات التفاضلية الخطية من الدرجة الأولى. لكى يتم ذلك سنقوم بإجراء تغيير على أسماء المتغيرات كما يلى:

$$u_1(x) = y(x)$$
$$u_2(x) = y'(x)$$
$$u_3(x) = z(x)$$
$$u_4(x) = z'(x)$$

وبالتالى يمكن كتابة نظام المعادلات من الدرجة الأولى كما يلى:

$u_1(a) = y_0$ حيث $u_1{}' = y'(x) = u_2$

$u_2(\mathrm{a}) = y_0^1$ حيث $u'_2 = y''(x) = f(x.u_1.u_2.u_3.u_4)$

$u_3(a) = z_0$ حيث $u_3{}' = z'(x) = u_4$

$u_4(\mathrm{a}) = z_0^1$ حيث $u'_4 = z''(x) = g(x.u_1.u_2.u_3.u_4)$

هذه المعادلات الأربعة تمثل نظام من المعادلات التفاضلية الخطية ذات الدرجة الأولى المكافىء لنظام المعادلتين (٩- ٢٧) و (٩- ٢٨) من الدرجة الثانية.

يمكن تعميم ذلك على أى معادلة تفاضلية من أى درجة حيث يمكن تحويل أى معادلة تفاضلية من الدرجة n مثلا إلى نظام مكون من عدد n من المعادلات التفاضلية التى كل منها من الدرجة الأولى.

يمكن وضع النظام السابق من المعادلات التفاضلية فى صورة متجهات على النحو التالى:

(٩-٢٩) $u'(x) = f(x)$ حيث القيم الابتدائية هى $u(a)=u_0$

فى المعادلة (٩- ٢٩) $u'(x)$ عبارة عن متجه كالتالى: $u'(x) = \begin{bmatrix} u'_1(x) \\ u'_2(x) \\ u'_3(x) \\ u'_4(x) \end{bmatrix}$

والطرف الأيمن عبارة عن متجه دوال كالتالى: $f(x) = \begin{bmatrix} u_2 \\ f(x.u_1.u_2.u_3.u_4) \\ u_4 \\ g(x.u_1.u_2.u_3.u_4) \end{bmatrix}$

كما أن القيم الابتدائية ستكون عبارة عن متجه كالتالى: $u_0 = \begin{bmatrix} y_0 \\ y_0^1 \\ z_0 \\ z_0^1 \end{bmatrix}$

وبذلك يمكن التعامل مع نظام المعادلات فى الصورة المتجهية السابقة ويمكن التعامل بهذه الصورة المتجهية فى ماتلاب كما سنرى فى الأمثلة التالية.

مثال ٩-٦

ضع المعادلة التفاضلية التالية فى صورة نظام من المعادلات التفاضلية المتجهية: $y'' - 3y' + 2y = 0$ حيث x>0 والقيم الابتدائية هى y(0)=1 و $y'(0) = 0$.

الخطوة الأولى: وضع المعادلة فى الصورة الرسمية للمعادلة التفاضلية كالتالى: $y'' = 3y' - 2y$

الخطوة الثانية: استخدام المتغيرات الجديدة كالتالى: u1(x)=y(x) و $u_2(x) = y'(x)$

وضع المتغيرات الجديدة فى صورة معادلات تفاضلية وستكون المعادلة التفاضلية الأولى كالتالى: $u'_1 = y'(x) = u_2(x)$ وستكون القيمة الابتدائية كالتالى: $u_1(0) = y(0) = 1$.

المعادلة التفاضلية الثانية ستكون: $u'_2 = y''(x) = 3y' - 2y$. وستكون القيمة الابتدائية: $u'_2(0) = y'(0) = 0$

وعلى ذلك فإن المعادلتين التفاضليتين ستكونان: $u'_1(x) = u_2(x)$ و $u'_2 = 3y' - 2y$ وهما بذلك يكونان نظام من المعادلات التفاضلية مكون من معادلتين يمكن وضعهما فى الصورة المتجهية التالية:

$$u(x) = \begin{bmatrix} u'_1(x) \\ u'_2(x) \end{bmatrix} = \begin{bmatrix} u_2(x) \\ 3u_2 - 2u_1 \end{bmatrix}$$

ومتجه القيم الابتدائية هو: $u_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ وسنرى بعد قليل كيفية استخدام ماتلاب لحل هذه الصورة من المعادلات.

٩-٦ حل المعادلات التفاضلية باستخدام ماتلاب

يوجد فى ماتلاب دالتين لحل المعادلات التفاضلية من الدرجة الأولى. الدالة الأولى هى ode23( ) وهى تستخدم طريقة رونج كوتا من الدرجة الثانية والدرجة الثالثة لحساب الحل. ماتلاب لديه طريقة للحساب الآلى لمقدار الخطوة h ولكى يتم ذلك فإنه يستخدم فى الحل درجتين الثانية والثالثة. الصورة العامة لهذه الدالة هى:

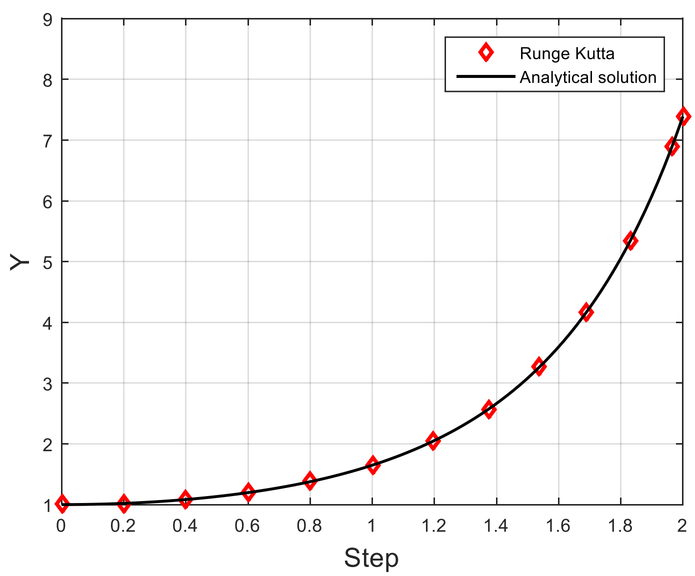
[xout,yout]=ode23('fun', span, y0)

حيث fun هى الدالة التى فى الطرف الأيمن من المعادلة التفاضلية حيث يمكن وضعها إما باستخدام الأمر inline( ) أو وضعها فى ملف دالة نصى واستخدام إسم هذا الملف. القيمة span تحدد مجال المتغير المستقل [a b] مثلا، و y_0 هى القيمة الابتدائية. الحل الناتج يعطى فى المتجهين y_{out} الذى يحتوى قيمة y(x) عند كل خطوة محددة فى المتجه

x_{out}. هناك أيضا فى ماتلاب الدالة ode45( ) التى تشبه تماما للدالة ode23( ) فيما عدا أنها تستخدم طريقة رونج كوتا من الدرجة الرابعة والخامسة وبالتالى فإنها ستكون أكثر دقة من الدالة ode23( ).

مثال ٩–٦

أوجد حل المعادلة التفاضلية $y'(x) = x * y$ حيث x فى المدى [0 2] والقيمة الابتدائية y(0)=1. لاحظ أن الحل التحليلى لهذه المعادلة التفاضلية هو $y(x) = e^{x^2/2}$. البرنامج التالى وشكل ٩–٧ يوضحان كل من الحل التقريبى باستخدام دالة ماتلاب ode23( ) والحل التحليلى حيث نلاحظ انطباقهما تقريبا.



شكل ٩–٧ الحل التقريبى باستخدام دالة ماتلاب ode23( ) والحل التحليلى

```
% using Runge Kutta
clear
f=inline('(x*y)');
[Xout, Yout]=ode23(f,[0 2],1);
curve=plot(Xout,Yout,'rd');grid;hold on; grid on;
set (curve,'LineWidth',2)
i=linspace(0,2);  % plotting real solution
y4=exp(i.^2/2);
plot(i,y4,'k-', 'linewidth',1.5)
ylabel('Y', 'fontsize',14)
xlabel('Step', 'fontsize',14)
axis([0 2 1 9]);
legend('Runge Kutta','Analytical solution')
```

هناك أيضا الدالة ( )ode113 وهى مثلها مثل الدالتين ( )ode23 و ( )ode45 سوى أنها تستخدم طريقة آدم Adam لحل المعادلات التفاضلية. وهذه الطريقة أكثر تعقيدا من طريقة رونج كوتا وهى طريقة متعددة الخطوات multistep ولذلك فإنها مكلفة جدا حسابيا. يوجد فى ماتلاب أيضا الدالتين ( )ode15s و ( )ode23s حيث الحرف s هنا يعنى stiff بمعنى جاسىء أو متيبس، مما يعنى أن هاتين الدالتين يستخدمان مع نوع معين من الدوال التفاضلية التى لا يكون لها حل تحليلى وتحتاج لخطوات صغيرة جدا مما يجعلها تحتاج لوقت أكبر للحصول على الحل.

مثال ٩-٧

استخدم ماتلاب لحل نظام لورينز للمعادلات التفاضلية المعرف بالمعادلات التالية:

$$x' = -\sigma x + \sigma y$$
$$y' = \rho x - y - xz$$
$$z' = -\beta z + xy$$

حيث الثوابت σ=10 و β=8/3 و ρ=28 والقيم الابتدائية هى x(0)=-8 و y(0)=8 و z(0)=27.

الخطوة الأولى هى وضع المعادلات الثلاثة فى الصورة المتجهية حيث سنفترض متجه x له ثلاث مركبات x(1)=x و x(2)=y و x(3)=z وبالتالى فإن المعادلات الثلاث السابقة يمكن كتابتها كالتالى:

$$x'(1) = -\sigma x(1) + \sigma x(2)$$
$$x'(2) = \rho x(1) - x(2) - x(1)x(3)$$
$$x'(3) = -\beta x(3) + x(1)x(2)$$

وعلى ذلك يمكن كتابة الصورة المتجهية التالية:

$$x' = \begin{bmatrix} x'(1) \\ x'(2) \\ x'(3) \end{bmatrix} = \begin{bmatrix} -\sigma x(1) + \sigma x(2) \\ \rho x(1) - x(2) - x(1)x(3) \\ -\beta x(3) + x(1)x(2) \end{bmatrix}$$

ومتجه القيم الابتدائية سيكون:

$$x'(0) = \begin{bmatrix} -8 \\ 8 \\ 27 \end{bmatrix}$$

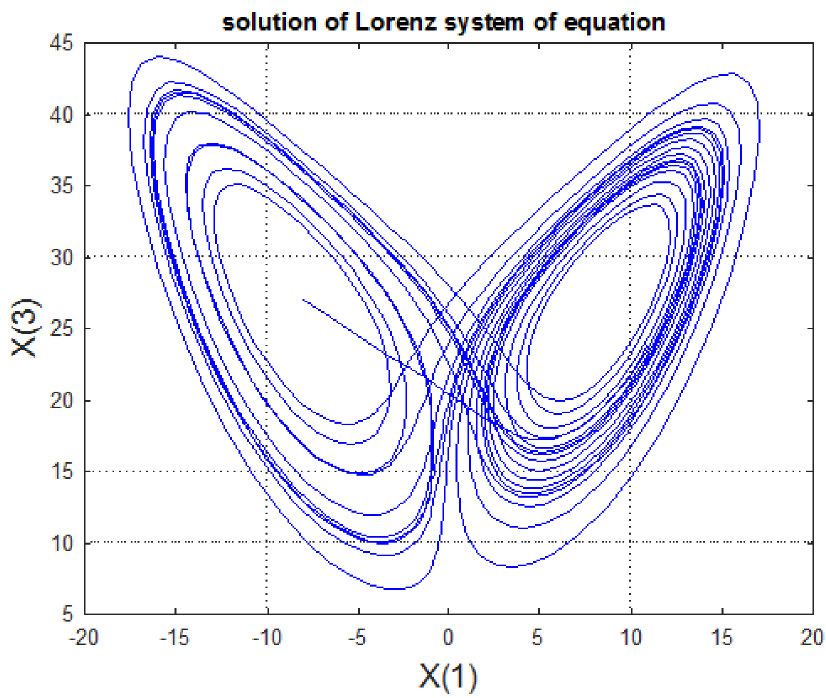
لكى نستخدم ماتلاب لحل هذا النظام من المعادلات فى الصورة المتجهية سنفتح ملف دالة ونسميه lorenz1 مثلا نكتب فيه متجه الدوال المكون من ثلاث صفوف كالتالى:

```
function xprime = lorenz1(t,x);
xprime=[-10*x(1) + 10*x(2);
    28*x(1) - x(2) - x(1)*x(3);
    -8/3*x(3) + x(1)*x(2)];
```

ثم البرنامج الأساسى الذى سيستخدم الدالة ( )ode45 للنداء على هذا الملف السابق ورسم النتيجة كالتالى:

```
%using matlab to solve system of 3 ODE
x0=[-8;8;27];
[t,x]=ode45(@lorenz1,[0,20],x0);
plot(x(:,1),x(:,3),'-b');grid on;
ylabel('X(3)', 'fontsize',14)
xlabel('X(1)', 'fontsize',14)
title('solution of Lorenz system of equation')
```

لاحظ أن ناتج هذا الحل سيكون مصفوفة مكونة من العديد من الصفوف على حسب عدد الخطوات التى سيضعها ماتلاب فى المدى [0 20] وأربع أعمدة. العمود الأول هو المتغير المستقل (الزمن t) والثلاث أعمدة التالية ستكون للمتغيرات المعتمدة x(1) و x(2) و x(3) حيث يمكن رسم أى اثنين من هذه المتغيرات وشكل ٩–٨ يوضح رسم x(1) مع x(3).



شكل ٩–٨ حل نظام معادلات لورينز المكون من ثلاث معادلات

مثال ٩–٨

استخدم دالة ماتلاب ode23( ) فى حل المعادلة التفاضلية التالية من الدرجة الثانية:

$$x^2y'' - xy' - 3y = x^2\log(x)$$

حيث: y(1)=-1 و $y'(1) = 0$ فى المدى [1 4] مع العلم أن الحل التحليلى هو: $y = \left(-\frac{1}{3}\log(x) - \frac{2}{9}\right)x^2 - \frac{7}{9x}$

الخطوة الأولى ستكون وضع المعادلة التفاضلية فى الصورة القياسية كالتالى: $y'' = \frac{y'}{x} + \frac{3y}{x^2} + \log(x)$

الخطوة الثانية هى استخدام متغيرات جديدة كالتالى:

$$u_1(x) = y(x) \quad \text{ومنها} \quad u_1(1) = -1$$

$$u_2(x) = y'(x) \quad \text{ومنها} \quad u_2(1) = 0$$

وبالتالى يمكن كتابة المعادلات التفاضلية من الدرجة الأولى كالتالى:

$$u'_1(x) = u_2(x)$$

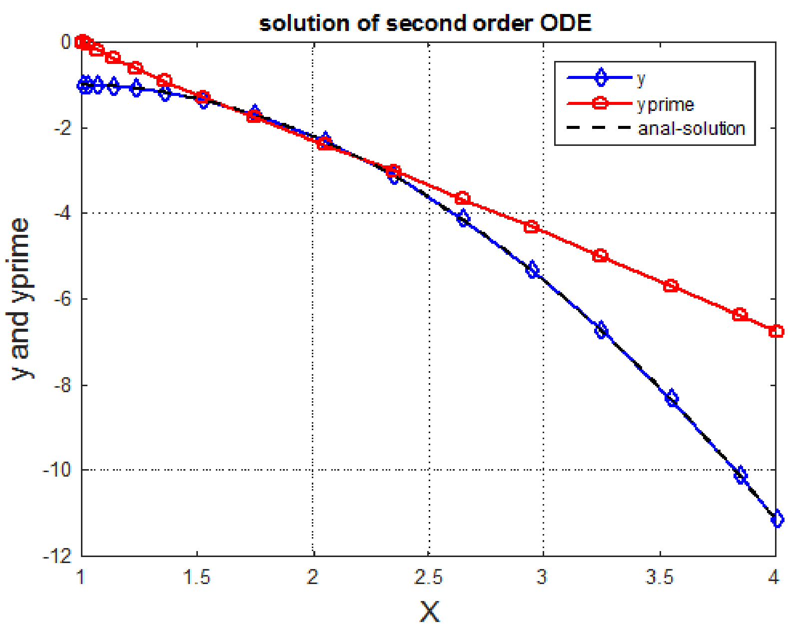
$$u'_2 = \frac{u_2}{x} + \frac{3u_1}{x^2} + \log(x)$$

وعلى ذلك ستكون الصورة المتجهية لهذا النظام من معادلتى الدرجة الأولى كما يلى:

$$u(x) = \begin{bmatrix} u'_1(x) \\ u'_2(x) \end{bmatrix} = \begin{bmatrix} u_2 \\ \frac{u_2(x)}{x} + \frac{3u_1}{x^2} + \log(x) \end{bmatrix}$$

ومتجه القيم الابتدائية سيكون: $u_0 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$

سنفتح ملف دالة نصية أسميناه ex7 يحتوى الدالة فى صورتها المتجهية كما يلى:



شكل ٩-٩ حل المعادلة التفاضلية من الدرجة الثانية فى المثال ٩-٨

```
function uprime = ex7(x,u);
uprime=[u(2);
        (u(2)/x)+(3*u(1)/x^2)+log(x)];
```

وسيكون البرنامج الذى ينادى على هذه الدالة لكى يحسب الحل التقريبي للمعادلة التفاضلية كما يلى:

```
%using matlab to solve second order ODE
clear
u0=[-1;0];
```

```
[x,u]=ode23(@ex7,[1,4],u0);
plot(x, u(:,1),'-bd','linewidth',1.5);hold on;grid on;
plot(x,u(:,2),'-ro','linewidth',1.5);hold on;
x=linspace(1,4);
anal_sol=x.^2.*(-log(x)./3 - 2/9) - 7./(9.*x);
plot(x,anal_sol,'--k','linewidth',1.5)
ylabel('y and yprime', 'fontsize',14)
xlabel('X', 'fontsize',14)
title('solution of second order ODE')
legend('y', 'yprime','anal-solution' )
```

شكل ٩-٩ يبين حل هذه المعادلة ومقارنته بالحل التحليلى. لاحظ أن كل من الحل التقريبي والتحليلى منطبقين (المنحنى الأزرق والأسود المتقطع)

٩-٧ تمارين

١- لديك المعادلة التفاضلية التالية $y' = e^{-x^2}$ مع القيمة الابتدائية y(0)=0. إحسب تقدير للقيم y(0.5) و y(1) و y(1.5) و y(2).

٢- حل المعادلة التفاضلية التالية: $x' = -xsin(t) + \sin(t)$ حيث x(0)=2+e في المدى [0 2π]،

أ- باستخدام طريقة أويلر مع عدد خطوات مقداره ٢ و ٥ و ١٠. إرسم الناتج مع عدد الخطوات.

ب-باستخدام طريقة النقطة المتوسطة مع عدد خطوات مقدارها ٥.

٣- أعد التمرين ٢ على المعادلة التفاضلية التالية: $y'(x) = -y(x) + 0.5e^{-x}$ حيث y(0)=0 في المدى [0 10]. ارسم النتيجة مع الخطوات باستخدام طريقة أويلر مع المقارنة بالحل التحليلى التالى: $y(x) = 0.5xe^{-x}$.

٤- حل المعادلات التفاضلية التالية بطريقة أويلر بمقدار خطوة h=0.2 و h=0.1 و h=0.05. إرسم الحل الناتج وقارنه بالحل التحليلى المعطى مع كل معادلة:

أ- $y'(x) = (\cos(y(x)))^2$ حيث y(0)=0 في المدى [0 10]، الحل التحليلى: $y(x) = tan^{-1}(x)$

ب- $y'(t) = \frac{1}{1+t^2} - 2[y(t)]^2$، y(0)=0، في المدى [0 10]، الحل: $y(t) = \frac{t}{1+t^2}$.

ت- $y'(t) = 0.25y(t)\left[1 - \frac{1}{20}y(t)\right]$، y(0)=1 في المدى [0 20]، الحل: $y(t) = \frac{20}{1+19e^{-t/4}}$

ث- $y'(x) = -[y(x)]^2$، y(1)=1 في المدى [1 10] ، الحل هو: y(x)=1/x.

ج- $y'(t) = (3t^2 + 1)y^2(t)$، y(0)=-1، المدى هو: [0 10]، الحل: $y(t) = -(t^3 + t + 1)^{-1}$

٥- حول المعادلات التفاضلية التالية ذات الدرجات العليا إلى نظم من معادلات الدرجة الأولى واستخدم ماتلاب لحل النظم الناتجة وقارن بالحل التحليلى المعطى:

أ- $y'''(t) + 4y''(t) + 5y'(t) + 2y(t) = 2t^2 + 10t + 8$، حيث y(0)=1 و $y'(0) = -1$، و $y''(0) = 3$، الحل التحليلى هو $y(t) = e^{-t} + t^2$.

ب- $y''(t) + 4y'(t) + 13y(t) = 40\cos(t)$، $y(0)=3$ و $y'(0) = 4$، والحل التحليلى هو:

$$y(t) = 3\cos(t) + \sin(t) + e^{-2t}\sin(3t)$$

٦- أعد التمرين ٤ مستخدما طريقة رونج كوتا من الدرجة الرابعة.

٧- أعد التمرين ٤ مستخدما دالتى ماتلاب ( )ode23 و ( )ede45.

الملحق ١

دوال ماتلاب المستخدمة فى هذا الكتاب

سنقدم فى هذا الملحق كل دوال ماتلاب المستخدمة فى هذا الكتاب مع شرح مبسط لكل منها وأمثلة بسيطة عليها لتكون بمثابة مرجع سريع لقارىء هذا الكتاب، وبالطبع فهذه ليست كل دوال ماتلاب لأن دوال ماتلاب أكثر من ذلك بكثير فى شتى المواضيع والتخصصات. عند كتابة أمثلة على هذه الدوال فى الجدول التالى سنتبع نفس التقليد الذى اتبعناه داخل الكتاب وهو أن استجابة ماتلاب لأى دالة ستكون باللون الأزرق.

الدالة	شرح مبسط للدالة

abs()	دالة تحسب القيمة المطلقة. abs(x) تحسب القيمة المطلقة للمتغير x. إذا كان المتغير x عبارة عن مصفوفة، فإن الدالة abs( ) تعطى القيمة المطلقة لكل عنصر فى المصفوفة. إذا كان المتغير x متغير مركب، فإن abs( ) يعطى مقدار هذا المتغير (الجذر التربيعى لمربع المركبة الحقيقية زائد مربع المركبة التخيلية). من أمثلة ذلك:
	`>> y=abs(-4)`
	`y =`
	`4`
	`>> x=[-2 1 -4];`
	`>> y=abs(x)`
	`y =`
	`2 1 4`
	`>> x=3+4i;`
	`>> y=abs(x)`
	`y =`
	`5`

sin()	دالة لحساب جيب أى زاوية بالتقدير الدائرى radian. sin(x) يعطى جيب الزاوية x على أن تكون x بالتقدير الدائرى. إذا كانت x مصفوفة فإن sin(x) تعطى جيب كل عنصر من عناصر المصفوفة على حدة. باقى الدوال المثلثية تعامل بنفس الطريقة. من أمثلة ذلك:
	`>> y=sin(pi/6)`
	`y =`
	`0.5000`

	>> x=[pi/6 pi/3 pi/2 pi]; >> y=sin(x) y = 0.5000 0.8660 1.0000 0.0000
sind()	حساب جيب أى زاوية بالدرجات. sind(x) تعطى جيب الزاوية x على أن تكون الزاوية x بالدرجات مثلها فى ذلك مثل الدالة sin(). باقى الدوال المثلثية تعامل بنفس الطريقة. سنعيد نفس أمثلة الدالة sin() ولكن مع كتابة الزوايا بالدرجات: >> y=sind(30) y = 0.5000 >> x=[30 60 90 180]; >> y=sind(x) y = 0.5000 0.8660 1.0000 0
asin()	الجيب العكسى، أى أن هذه الدالة تعطى الزاوية بالتقدير الدائرى التى جيبها هو معامل هذه الدالة، فمثلا asin(0.5) سيعطى الزاوية بالتقدير الدائرى التى جيبها يساوى 0.5. باقى الدوال المثلثية تعامل بنفس الطريقة. ومن أمثلة ذلك: >> x=[0.5 0.8660 1 0]; >> y=asin(x) y = 0.5236 1.0471 1.5708 0 لاحظ أن: >> [pi/6 pi/3 pi/2 0] ans = 0.5236 1.0472 1.5708 0
asind()	مثلها مثل الدالة asin(x) سوى أنها تعطى الزاوية بالدرجات بدلا من التقدير الدائرى. باقى الدوال المثلثية تعامل بنفس الطريقة. ومن أمثلة ذلك: >> x=[0.5 0.8660 1 0]; >> y=asind(x) y = 30.0000 59.9971 90.0000 0

log()	اللوغاريت الطبيعى. Log(x) يحسب اللوغاريتم الطبيعى natural logarithm للأساس e للمتغير x. هناك أيضا log10(x) التى تحسب لوغاريتم المتغير x للأساس 10 و log2(x) الذى يحسب لوغاريتم x للأساس 2. من أمثلة ذلك: >> log(2) ans = 0.6931 >> log10(2) ans = 0.3010 >> log2(2) ans = 1
sqrt()	حساب الجذر التربيعى. sqrt(x) تحسب الجذر التربيعى للمعامل x. من أمثلة ذلك: >> x=[2 4 6]; >> y=sqrt(x) y = 1.4142 2.0000 2.4495
ceil()	التقريب لأعلى رقم صحيح. ceil(x) تقريب x لأعلى رقم صحيح، كمثال على ذلك: >> y=ceil(2.1) y = 3
floor()	التقريب لأقل رقم صحيح. floor(x) تقرب x لأقل رقم صحيح، كمثال على ذلك: >> y=floor(2.9) y = 2
round()	التقريب لأقرب رقم صحيح. round(x) تقريب x لأقرب رقم صحيح، من أمثلة ذلك: >> y=round(2.6) y = 3 >> y=round(2.3) y = 2

	>> y=round(2.5) y = 3
rem()	دالة حساب الباقى من القسمة، rem(x,y) تقسم x على y وتعطى الباقى من هذه القسمة، وكمثال على ذلك: >> y=rem(8,5) y = 3
exp()	حساب e أس معامل الدالة، exp(x) تحسب e^x، كمثال على ذلك: >> y=exp(1) y = 2.718281828459046
format	طريقة عرض النتيجة، format long يعرض النتيجة فى صورة رقم صحيح من ١٦ خانة، بينما format short فيعرض النتيجة فى خمس خانات فقط. تذكر أن النتيجة تعرض فى هذه الحالة فى خمس خانات ولكن ماتلاب يكون محتفظ بها فى صورة ١٦ خانة ويعرض فقط خمس خانات، من أمثلة ذلك: >> format long >> y=exp(1) y = 2.718281828459046 >> format short >> y=exp(1) y = 2.7183
clc	تمسح أو تنظف شاشة نافذة الأوامر. ويبدأ دليل الكتابة من أعلى يسار الشاشة. لاحظ أن جميع المتغيرات فى أى جلسة عمل تظل محتفظة بقيمتها بعد الأمر clc.
clear	الأمر clear يصفر جميع المتغيرات المستخدمة فى نافذة الأوامر من أول جلسة العمل، clear x يصفر المتغير x فقط فى نافذة الأوامر.

plot()	أمر الرسم plot() له العديد من الصور والعديد من الدوال المصاحبة وننصح باللجوء إلى مساعدة ماتلاب التى تعطى الكثير من المعلومات عنه عن طريق كتابة help plot. ولكننا سنعرض هنا الأوامر التى ترسم الدالتين sin() و cos() لاستخدامها كمرجع سريع لدوال الرسم كثيرة الاستخدام. x=0,0.1,10; y=sin(x); z=cos(x); plot(x,y,'r+:',x,z,'b>:'); grid on; title('plotting two functions'); ylabel('sin(x), cos(x)'; xlabel('x in radians'); legend('sin(x)', 'cos(x)'); الأمر plot() يرسم المنحنى الأول x,y باللون الأحمر r و : تعنى منحنى منقط مع استخدام + كنقط رسم. والمنحنى الثانى x,z باللون الأزرق b المنقط أيضا واستخدام مثلث > كنقط للرسم. حاول تنفيذ هذه الأوامر فى ماتلاب لترى الشكل الناتج.
tic...toc	الأمر tic يبدأ ساعة إيقاف تظل تعمل حتى الأمر t=toc حيث يتم تخزين الزمن الذى مر فى ساعة الإيقاف فى المتغير t. يتم استخدام الأمرين tic و toc لحساب زمن تنفيذ مجموعة الأوامر المحصورة بينهما.
diff()	إجراء التفاضل على تعبير رمزى وهناك صندوق أدوات كامل للحساب الرمزى، وكمثال على ذلك: syms x; y=sin(5*x); f=diff(y) f = 5*cos(5*x)
int()	إجراء التكامل على تعبير رمزى، كمثال على ذلك: syms x; y=x^2; f=int(y) f=int(y) f = x^3/3
trace()	أثر المصفوفة هو مجموع عناصر قطرها الرئيسى $a_{11}+a_{22}+\ldots+a_{nn}$ كمثال على ذلك: >> A=[1 2 3;4 5 6;7 8 9] A = 1 2 3 4 5 6 7 8 9 >> x=trace(A) x = 15

zeros()	الحصول على مصفوفة كل عناصرها أصفار بأى أبعاد، كمثال على ذلك zeros(2,3) تعطى مصفوفة من صفين وثلاث أعمدة كل عناصرها أصفار.
	`>> x=zeros(2,3)`
	`x =`
	`0 0 0`
	`0 0 0`

ones()	الحصول على مصفوفة كل عناصرها وحايد بأى أبعاد، كمثال على ذلك ones(2,3) تعطى مصفوفة من صفين وثلاث أعمدة كل عناصرها وحايد.
	`>> x=ones(2,3)`
	`x =`
	`1 1 1`
	`1 1 1`

transpose	دوران المصفوفة، بجعل الأعمدة صفوف وجعل الصفوف أعمدة، وتكتب فى ماتلاب هكذا B=A’ أى جعل المصفوفة B تساوى دوران المصفوفة A، كمثال على ذلك:
	`A =`
	`1 2`
	`3 4`
	`>> B=A'`
	`B =`
	`1 3`
	`2 4`

inv()	معكوس المصفوفة. بفرض أن لدينا المصفوفة A فإن معكوس A يكون هو المصفوفة التى لو ضربت فى A لأعطت مصفوفة الوحدة. هناك شرط لوجود معكوس لأى مصفوفة هو أن تكون محددة هذه المصفوفة لا تساوى صفر.
	`A =`
	`1 2`
	`3 4`
	`>> B=inv(A)`
	`B =`
	`-2.0000 1.0000`
	`1.5000 -0.5000`

det()	حساب المحددة لأى مصفوفة، كمثال على ذلك det(A) يعطى قيمة محددة المصفوفة A: A = 1 2 3 4 >> x=det(A) x = -2
linespace()	الصورة العامة لهذا الأمر هى linespace(x1,x2,N) حيث يقوم بتقسيم المدى من x1 حتى x2 إلى عدد N من المسافات المتساوية، وفى حالة عدم ذكر عدد نقاط التقسيم N تكون القيمة التلقائية التى يأخذها ماتلاب هى 100.
max()	الصورة العامة لهذه الدالة هى [max m]=max(x) حيث يتم حساب أكبر قيمة فى المتجه x ووضعها فى المتغير max ورتبة العنصر الذى يحتوى هذه القيمة العظمى توضع فى المتغير m. كمثال على ذلك: >> x=[1 5 3]; >> [max m]=max(x) max = 5 m = 2
linsolve()	تحسب حل مجموعة من المعادلات الخطية على الصورة Ax=b، باستخدام تحليل المصفوفة A إلى مصفوفتين مثلثة علوية ومثلثة سفلية، والصورة العامة لهذه الدالة هى x=linsolve(A,b) حيث x هى متجه الحل و A هى مصفوفة المعاملات و b هى مصفوفة الثوابت. كمثال على ذلك: >> A=[1 -1 2 -1 ;2 -2 3 -3 ; 1 1 1 0 ;1 -1 4 3]; b=[-8;-20;-2;4]; >> X = linsolve(A,b) X = -7.0000 3.0000 2.0000 2.0000

<table>
<tr><td>interp1()</td><td>الصورة العامة لهذه الدالة هى: interp1(x,y,'linear')، فإذا كان كل من x و y متجهين لهما نفس الطول فإن الدالة ستقوم بعمل استيفاء لنقاط المتجهين فى صورة منحنى قطع خطية تمر بجميع هذه النقاط. هناك اختيارات أخرى غير الاختيار 'linear' يمكن استخدامها ومنها مثلا الاختيار 'spline' الذى يقوم بعمل الاستيفاء فى صورة منحنى من الدرجة الثالثة يمر بكل نقطتين متتاليتين. الدالة y_new=interp1(x,y,x_new,'linear') يقوم بعمل استيفاء بمنحنى خطى لنقاط المتجهين x و y ويحسب من هذا الاستيفاء النقاط الجديدة y_new المقابلة للنقاط x_new.
استدعى المساعدة من ماتلاب وانظر الفصل ٧ لترى أمثلة على ذلك.</td></tr>
<tr><td>vander()</td><td>دالة تعطى مصفوفة فاندرموند Vandermond لأى متجه x. تستخدم مصفوفة فاندرموند فى الاستيفاء بكثيرات الحدود أو حل المعادلات الخطية، الصورة العامة لهذه الدالة هى: vander(x) وكمثال على ذلك:
>> x=[1 2 3];
>> v=vander(x)
v =
1 1 1
4 2 1
9 3 1</td></tr>
<tr><td>fliplr()</td><td>عكس مصفوفة أو متجع من اليسار لليمين، الصورة العامة لهذه الدالة هى fliplr(x) حيث x متجه أو مصفوفة، ومن أمثلة ذلك:
v =
1 1 1
4 2 1
9 3 1
>> R=fliplr(v)
R =
1 1 1
1 2 4
1 3 9</td></tr>
</table>

polyfit()	إيجاد كثيرة حدود من الدرجة n تستوفى نقاط البيانات x و y، والصورة العامة لهذه الدالة هى: p=polyfit(x,y,n)، الناتج p هو متجه عدد عناصره n+1 يحتوى معاملات كثيرة الحدود متناقصة الأس. من المفضل أن تكون n أقل من عدد نقاط البيانات. كمثال على ذلك:
	`>> x = [-2 -1 1 2];`
	`>> y = [10 4 6 3];`
	`>> p=polyfit(x,y,2)`
	`p =`
	`   0.5000   -1.2000    4.5000`
	وهذا يعنى أن كثيرة الحدود $0.5x^2-1.2x+4.5$ ستكون هى كثيرة الحدود من الدرجة الثانية التى تستوفى نقاط البيانات x و y.

polyval()	دالة تحسب قيمة كثيرة الحدود عند قيمة معينة للمتغير x، والصورة العامة لها هى: y=polyval(p,x) حيث يتم حساب قيمة كثيرة الحدود التى معاملاتها هى المتجه p الذى يتكون من n+1 من العناصر، عند القيمة x، (أنظر الدالة polyfit())، كمثال على ذلك:
	`>> x = [-2 -1 1 2];`
	`>> y = [10 4 6 3];`
	`>> p=polyfit(x,y,4);`
	`>> y1=polyval(p,-1)`
	`y1 =`
	`    4`

spline()	دالة سبلاين للاستيفاء، والصورة العامة هى: pp=spline(x,y) حيث تعطى استيفاء لنقاط البيانات x و y فى صورة مقاطع كثيرات حدود من الدرجة الثالثة، كمثال على ذلك:
	`>> x = [0 1 2 3 4];`
	`>> y = [0 1 8 27 64];`
	`>> pp = spline(x,y);`

ppval()	الصورة العامة لهذه الدالة هى v=ppval(pp,x) حيث يتم التعويض بقيمة x فى دالة الاستيفاء التى pp التى هى عبارة عن مقاطع كثيرات حدود (أنظر الدالة spline())، كمثال على ذلك:
	`>> x = [0 1 2 3 4];`
	`>> y = [0 1 8 27 64];`
	`>> pp = spline(x,y);`
	`>> v=ppval(pp,4)`

	v = 64
diff()	التفاضل العددى، والصورة العامة لهذه الدالة هى y=diff(x) حيث المتجه y يمثل الفرق بين كل قيمتين متتاليتين للمتجه x. كمثال على ذلك: >> x=[1 2 3 4 5]; >> y=diff(x) y = 1 1 1 1 يجب أن نفرق بين هذا التفاضل، ودالة تفاضل أخرى diff() يتم إجراؤها على المتغيرات الرمزية حيث فى هذه الحالة يتم حساب التفاضل الرمزى.
trapz()	التكامل العددى باستخدام قاعدة شبه المنحرف، والصورة العامة لهذه الدالة y=trapz(x) حيث سيكون المتجه y يساوى تكامل المتجه x، وكمثال على ذلك: >> X = 0:pi/100:pi; >> Y = sin(X); >> Z = pi/100*trapz(Y) Z = 1.9998 هناك صورة عامة أخرى وهى z=trapz(x,y) حيث سيكون المتجه z يساوى التكامل العددى للمتجه y بالنسبة للمتجه x، ومن أمثلة ذلك: >> X = 0:pi/100:pi; >>Y = sin(X); >> Z = trapz(X,Y) Z = 1.9998
quad()	التكامل العددى باستخدام قانون سمبسون Simpson، والصورة العامة لهذه الدالة هى: y=quad('fun',a,b) حيث يتم تكامل الدالة fun عدديا فى المدى من a حتى b ووضع النتيجة فى y. تقوم هذه الدالة بتقسيم المدى [a b] إلى قسمين، ثم ثلاثة، ثم أربعة، وهكذا، وفى كل مرة يتم حساب التكامل العددى باستخدام قانون سمبسون وتتوقف عملية التكرار هذه عندما يكون الفرق بين محاولتين أقل من أو يساوى $1x10^{-6}$. كمثال على ذلك: >> quad('exp(x)',0,2)

	ans = 6.3891 هناك الصورة العامة التالية أيضا التى تعطى عدد مرات التكرار اللازمة للوصول للخطأ $1x10^{-6}$ كما فى المثال التالى: >> [I,n]=quad('exp(x)',0,2) I = 6.3891 n = 29 مما يعنى أن التكامل تم بعد ٢٩ محاولة.
quad2d()	التكامل العددى ثنائى الأبعاد، والصورة العامة هى: I=quad2d('f(x,y)'a,b,c,d) حيث يتم التكامل العددى للدالة f(x,y) مع تغير x فى المدى [a b] وتغير y فى المدى [c d]. أنظر الدالة quad().

ode23()	دالة تستخدم طريقة رونج كوتا من الدرجة الثانية والثالثة لحل المعادلات التفاضلية، الصورة العامة لهذه الدالة هى: [xout,yout]=ode23('fun',span,y0)، حيث fun هى الطرف الأيمن من المعادلة التفاضلية بحيث يحتوى الطرف الأيسر على عنصر أعلى درجة فقط للمعادلة التفاضلية، و span هو المدى الذى سيتم خلاله حل المعادلة التفاضلية، y0 تمثل القيم الابتدائية للمعادلة التفاضلية، الحل الناتج يعطى فى yout التى تمثل y(x) كدالة فى x المعطاة فى xout، وكمثال على ذلك حل المعادلة التفاضلية $y'(x) = x * y$ الذى يمكن كتابته كما يلى:
	`f=inline('(x*y)');`
	`[xout, yout]=ode23(f,[0 2],1);`
	حيث يمكنك رسم حل المعادلة الناتج yout و xout ومقارنته بالحل التحليلى أو الصحيح لهذه المعادلة وهو: $y(x) = e^{x^2/2}$.
	هناك أيضا الدالة ode45() التى تحل المعادلات التفاضلية باستخدام طريقة رونج كوتا من الدرجة الرابعة والخامسة وهى تشبه تماما الدالة ode23().