Mathematica, Packs for Mechanical Engineers – and the elastic torsion problem

C. Cetinkaya
Wolfram Research, Inc., Champaign, Illinois, USA.
`cetin@wri.com`

G. Keady
Department of Mathematics,
University of Western Australia
`keady@maths.uwa.edu.au`

A. Triulzi
School of Mathematical Sciences,
Queen Mary and Westfield College, University of London
`arrigo@maths.qmw.ac.uk`

16 Dec 1996

Abstract

Mathematica does nearly all the ordinary engineering mathematics which working engineers need. *Mathematica* has always done more than Computer Algebra (CA). Numerics, graphics, and user-interface matters have been smoothly integrated with the CA. *Mathematica* 3 adds mathematical word processing and hyperlinks. This opens the way to truly impressive, useful 'electronic books', 'interactive engineering handbooks', etc..

The ratio of engineering users of *Mathematica* to academic mathematician users is already greater than one, and rapidly increasing. The main part of the paper reports on a Pack - suite of useful code - for Mechanical Engineers. The Pack - an 'interactive engineering handbook' - described in the paper is incontrovertibly useful for engineers, but, as this is a conference for mathematicians, the thrust of the paper concerns the relevance of the Pack for applied mathematicians

– both for our teaching of engineering mathematics and for our research. To illustrate matters concerning the latter, the elastic torsion problem is used as an example, and some developments from earlier work are reported.

1 Introduction

Wolfram Research are about to release an Elastic Systems Pack, written in *Mathematica* 3, for Mechanical Engineer users. The first author (CC) wrote this Pack. A general description is given in the Appendix to this paper. GK wrote the main part of the paper which, in part, can be considered an independent review of (the elastic torsion) part of the Pack. The Appendix is by CC.

This is a conference on computer *technology in mathematics*. CAS - Computer Algebra Systems - in general, and *Mathematica* in particular, are emerging as mass-market software products for 'users' of mathematics. This paper illustrates some points related to this.

There will be implications, for applied mathematicians, of the technology associated with this Pack, and the following more general items:
• mass-market CAS technology, where engineering users are the largest single group of users (see §1, §6),
• university teaching of undergraduate engineering mathematics (see §1.1.1),
• applied mathematical research at the interface with engineering (see §2-).
These points will be treated, approximately in the order given above, in this paper.

There is a little applied-mathematical research content in this paper. This concerns the elastic torsion problem. This problem is just one of very many topics treated in the Pack. The second author (GK) has long-term research interests in the torsion problem, both concerning qualitative properties of solutions and the collection of exact solutions for particular domains. The paper [KA] (available on the Web) describes a continuing project. See §2.1 for how this paper differs from the earlier one.

The abbreviation 'CAS' is often used because it is shorter to write than *Mathematica*. We use 'CAS' when the CA capabilities – rather than numerics, graphics, etc. – are essential. Sometimes 'CAS' is used where CA Systems besides *Mathematica* could be used, though we stress that no other CA System has, as yet, Engineering Packs of the quality of *Mathematica*'s.

1.1 Engineering Packs

Participants at this Conference are already aware that *Mathematica* 3 combines, in a well-integrated and coherent manner, CA, numerics, graphics, mathematical word-processing and hyperlinks. The promise for 'Electronic Books' of applied-mathematical and engineering data is clear. It is still very early in the history of developing these, so substantial examples realising the promise are scarce. The Electrical Engineering Pack from WRI was an early example (1994). There is also a third-party Mechanical Systems Pack (1995). A review of relevant Packs at mid 1995 is given in [Ce]. The preceding predate the major new facilities in *Mathematica* 3.

This paper and [KA] suggest that it is via Engineering Packs that many, perhaps most, engineers - in the very near future - will look up their tables of engineering data. The Appendix of this paper, describing the Elastic Systems Pack, is further evidence for this.

1.1.1 Undergraduate engineering maths education

Even before the advent of publication of engineering data in CAS, CAS were successful in providing engineering undergraduates (as well as practising engineers) with tools that they were able to use in helping them with their mathematically-formulated engineering problems. In some universities, *Mathematica* is used by the engineering lecturers in sufficiently advanced engineering courses. Turning now to engineering-mathematics courses, the paper [KFW] contains some observations, made at the time when Student Versions of the CAS were beginning to be used by a significant fraction of the students. (A Web-browsable version of the paper is at GK's Web pages. Papers like this become dated very quickly. Delivery of CAS materials through the Web is not mentioned in that paper, and nor were Engineering Packs.) Different styles of CAS-in-teaching are possible: one of the widely-used courses [DPU] has *Mathematica* as a central part, and is in use at NUS, Singapore.

Useful Engineering Packs will increase further the uptake of *Mathematica* by engineers. Not least amongst the implications for academic mathematicians are those concerning teaching. If the engineering academics at University U are using (mathematical) Package P in their engineering teaching, mathematicians at U teaching Engineering Mathematics classes to the same students will be expected, by their engineering colleagues, to do their best to make educationally effective use of Package P. (Caveats are that the Engineering Mathematics course is sufficiently advanced, the students are sufficiently able, and so on.) In universities where the mathematicians are ahead of their engineering lecturer colleagues in using CAS with engineering maths students, the presence of well-written, commercially-produced, widely-used, 'electronic books of engineering data' – like the *Mathematica* Packs – will help motivate *all* of the students. *All* of the students includes those who have not, prior to this, totally trusted the idea that CAS were for engineers too, and not just for mathematicians. Engineering students will appreciate CAS more if they see it as a real tool that a significant proportion of them will use in their careers.

The *Mathematica* materials which are already available accompanying widely selling Engineering and Engineering Mathematics text books (e.g. [ON, Zi, Vv, MH]) will develop in ways which harmonise with Packs. The same *Mathematica* facilities, as used in the Packs, are useful for the accompaniments to text-books too.

1.1.2 Practising and research engineers

The market for *Mathematica*'s Engineering Packs is the engineering community. Practising engineers will typically look up data in the interactive handbooks. Research engineers will also have the skills to add functionality following the style of, and examples in, the Pack. When, as reported below, GK adds code for a different shape (e.g. lenses), or develops some alternative representation (e.g. that for the sector's torsion function given in §3.2.1), he is using skills supposed of a research engineer.

It should be added that, at present, there is much greater usage of *Mathematica* by research engineers than practising ones. The various Engineering Packs are expected to spread the use of *Mathematica* widely through the engineering community.

2 The Elastic Torsion Problem

Let Ω be a plane domain. The elastic torsion problem is to find a function u whose Laplacian is constant in Ω - taken to be minus one in this paper -, with $u = 0$ on the boundary of Ω:

$$-\Delta u = 1 \quad \text{in} \quad \Omega, \qquad\qquad u = 0 \quad \text{on} \quad \partial\Omega = \text{boundary}(\Omega). \tag{1}$$

The quantity

$$S = \int_\Omega u, \tag{2}$$

is called the *torsional rigidity* and measures the 'overall resistance to twisting' of a bar of cross-section Ω.

When Ω is a disk centred at the origin, and with radius a,

$$u = \frac{1}{4}(a^2 - r^2) = \frac{1}{4}(a^2 - x^2 - y^2), \tag{3}$$

when written in polar and Cartesian coordinates respectively. There are several other geometrically specified domains which, using Cartesian coordinates, have polynomial or rational function torsion functions. (See the Pack for the equilateral triangle: u is then a cubic polynomial. See [KA] for certain 'lunes' for which u is a very simple rational function.) In general, though, given a shape Ω, finding closed-form formulae for its torsion function u and its torsional rigidity S is a nontrivial task. Indeed explicit exact solutions are the exception, and, if one were to choose shapes 'at random', typically numeric solution would be required. However, shapes used for beams in engineering practice are not 'chosen at random', and, with an appropriately large engineering mathematical armoury, substantial progress has been made at exact solution. Series solution, different coordinate systems, special functions and complex variable techniques have all been used for the elastic torsion problem. These are all techniques excellently handled by *Mathematica*. The broad-brush picture is 'if the exact solution gets messy, solve numerically': but note that exact solutions in special cases retain a role in checking the numerical schemes. See §4.3 and the Appendix, for routes through to numerics.

We remark that an accurate and comprehensible account of the torsion problem, written for engineers, is given in Chapter 4 of the Pack.

We use, §3.2.3 and §4.1.3, the variational characterisation of S ([PS, So]). Define, for nonconstant functions v,

$$S_{LB}(v) = \frac{(\int_\Omega v)^2}{\int_\Omega |\nabla v|^2}. \tag{4}$$

The torsion function u - and scalar multiples of it - satisfy

$$\frac{1}{S_{LB}(u)} = \min_v \frac{1}{S_{LB}(v)}, \tag{5}$$

where the minimisation is done over functions in $W_2^1(\Omega)$ vanishing on the boundary of Ω. We will use this to find rigorous bounds on S for various shapes.

There are also a number of useful integral identities concerning S, some helpful for explicit calculations for certain shapes: see [KA, KM].

2.1 How this paper differs from the AEMC96 paper

(A Web-browsable html version of the AEMC96 paper [KA] is up at GK's homepage. The paper [KA] was written - and reproduced in the Conference Proceedings - in *Mathematica* 3-β to emphasise that the technology was coming. However, mathematicians world-wide have now seen *Mathematica* 3 and there is no need to make that point again. We have docilely followed the ATCM97 guidelines. Tools for *Mathematica* 3 to LaTeX translation are in preparation, but we did not use them.)

There are substantial differences concerning the treatment both of the elastic torsion problem, and of the Pack. When [KA] was written, the release of the Pack was too far in the future to review its contents. The focus in [KA] was on some additional torsion functions for shapes not in the Pack (certain lens-shaped domains) and new results for them. This paper is different in that we now can review the torsion function chapter of the Pack, and investigate in §3 how its basic solutions can be developed. There is, though, a small part of this paper, §4, in which we report some developments concerning solutions in lens-shaped domains made in the year since [KA] was accepted. This part of this paper should be read in tandem with [KA].

The overall focuses of the two meetings were different. In [KA] the goal was to report new applied-mathematical results related to Engineering Mathematics, and the 'technology' was a sub-theme only. In this meeting, the technology - *Mathematica* and its new Packs - is important, and the fact that, for the elastic torsion problem, new results can be found - using CAS - is merely an illustration that applied mathematicians can interact usefully with the new technology. We also now allow ourselves to give a subsection near the end of the paper, §5, on pure-mathematical uses of CAS related to the torsion problem.

3 Cross-sections Ω in the Pack

The Elastic Systems Pack includes the main shapes in the present books of engineering tables. Specific shapes included are circle, ellipse, equilateral triangle, rectangle, sector - and the special case of a semicircle.

Mostly the mathematics in the Pack is deliberately kept at the level that present-day working engineers are able to check.

In the Pack, there is a lot of useful graphics, making it both more interesting and more easily comprehended than the book treatments of the subject.

3.1 Basic shapes

The Pack favours familiar results over exotic ones. In a few rare instances where short formulae which *Mathematica* can evaluate simply are available, the Pack goes beyond the usual engineering books of tables. (See §3.2.1 for the torsional rigidity of a sector.)

The torsion functions for the rectangle and for the sector are presented through Fourier series engineers know from separation-of-variables techniques.

3.2 Basic shapes, sector

3.2.1 Closed form solution

The Pack has been written to be accessible to its engineering users. Further research with CAS technology as a tool enables more to be done than is currently in the Pack. (Design decisions to keep the Pack simple and accessible mean that there is room for incremental developments in later editions, or published independently.)

In 1995, GK called to CC's notice, the 1880s formula (in terms of PolyGamma functions) for the torsional rigidity for the sector. This is now in the Pack, and is an example of providing to a wider audience an old result, formerly known only to specialists. In late 1996, GK asked CAS assistance to sum the Fourier series representation of the torsion function for the sector. None of the authors knew if there would be a closed-form solution, in spite of some of us having read widely on the problem. The Fourier series for the torsion function itself can be summed exactly in terms of HyperGeometricPFQ ${}_pF_q$ functions. (The CAS was helped by replacing $r^{j/(2t)}\cos(j\theta/(2t))$ terms by $\mathrm{Re}(Z^j)$ ... but that was the only help provided to it.) Consider a sector with angle $2t\pi$, and axis of symmetry $\theta = 0$. Then

$$\begin{aligned} u_{\mathrm{sector}} &= -\frac{r^2}{4}\left(1-\frac{\cos(2\,\theta)}{\cos(2\,t\pi)}\right) \\ &\quad +\frac{16a^2t^2}{\pi}\sum_{j=1}^{\infty}\frac{(-1)^j\left(\frac{r}{a}\right)^{\frac{2j-1}{2t}}\cos(\frac{(2j-1)\theta}{2t})}{(2j-1)\,(2j-1+4\,t)\,(2j-1-4\,t)}\,. \end{aligned} \tag{6}$$

Below, $\mathrm{Re}(Z) = ((r/a)^{1/(2t)})\cos(\theta/(2t))$. We have $u_{\mathrm{sector}} = \mathrm{Re}(w_{\mathrm{sector}})$ where

$$\begin{aligned} w_{\mathrm{sector}} &= -\frac{r^2}{4}\left(1-\frac{a^2Z^{4\,t}}{r^2\cos(2\,t\pi)}\right) \\ &\quad -\frac{16a^2t^2Z}{\pi}\frac{{}_4F_3(\frac{1+4t}{2},\frac{1-4t}{2},1,\frac{1}{2};\frac{3+4t}{2},\frac{3-4t}{2},\frac{3}{2};-Z^2)}{(1+4\,t)\,(1-4\,t)}\,. \end{aligned} \tag{7}$$

Simplify, given the right parameter to say the sector is a semicircle, transforms the ${}_4F_3$ functions into ${}_2F_1$ ones - correctly and so that, with a little checking, it can be seen that they give the $\log((1-z)/(1+z))$ like terms in the familiar elementary function representation of the semicircle torsion function.

After the Pack is published, GK hopes to provide (via GK's Web pages), the code for the ${}_pF_q$ solutions. One way that the material might be developed, after it is tested, to harmonise with the Pack, is to provide amongst optional arguments for TorsionFunction, etc,

`Method -> HypergeometricPFQSolution`, or
`Method -> TruncatedFourierSeriesSolution`.

(We remark that the sector solutions - as series and as ${}_pF_q$ forms - generalise from the plane case to spherical cones in R^N, N-dimensions. The N appears in the parameters of the ${}_pF_q$ solution.)

There are other sectors besides the semicircle where closed-form elementary function torsion functions are available. The quadrant, and the shaft split along a radius, are examples. For these, a

`Method -> ElementaryFunctionSolution`

might be provided. A research engineer wanting to provide code for these would merely take the exact solution, say from [Lo], and edit the semicircle solution from the Pack into the desired form.

3.2.2 Further series solutions

The series solution for the sector, given in the Pack, is what would be generalised to treat 'sectorial rectangles': see [Lo], p319. A special case which is deemed to be sufficiently important to be in 'Roark's Formulas' [YR] is the split hollow shaft. These and more can be obtained by straightforward development of the code in the Pack.

It may be worth commenting that shapes like that of a shaft split along a radius are examples where straightforward numerical computation can be relatively difficult. Thus exact solution, as in the Pack, has its use. ([LSU] use bipolar coordinates to give series solutions to the case when the disk is cut, from the boundary, only part way along a radial line. It happens that some exact solution work based on their series occuring within the series for the torsional rigidity can be done. Again ${}_pF_q$ functions arise.)

3.2.3 Variational bounds

Of course, finding variational approximations to S for shapes where S is exactly known is a test of the method only. To give a simple example from one of the Basic Shapes of the Pack, consider for Ω the semicircle in the upper half-plane, radius 1. For this semicircle, substitute the trial function

$$v_{SC} = \frac{1}{2} y(\sqrt{1-x^2} - y), \tag{8}$$

in (4). $S_{LB}(v_{SC})$ is very easily found exactly and differs from the exact value $\pi/8 - 1/\pi$ by just 1%.

More generally, consider shapes Ω which can be described by

$$\Omega = \{(x,y) \mid Y_m(x) < y < Y_p(x)\ \}. \tag{9}$$

and trial functions

$$v = \frac{1}{2}(y - Y_m(x))(Y_p(x) - y). \tag{10}$$

(The trial function solves the Dirichlet problem $v = 0$ on $\partial\Omega$, but with $-\Delta v = 1$ replaced by $-v_{yy} = 1$.) This approximation is already used by engineers for long, slender, slowly-varying shapes, a topic that is related to that in §3.3. (For the ellipse shape from the Pack, (10) substituted into (4) gives the exact S.) With the reassurance that, for shapes in the Pack, we could obtain good bounds by this method, we applied it to other shapes for which, to date, we do not have closed-form torsional rigidities. The bounds, presented in §4.1.3, are close to S.

The summary is that, with CAS to do the routine integrals, and some exact solutions already there – in the Pack – against which to check, illustration of the bounds is easy and enjoyable.

3.3 Thin regions

The torsional rigidity of a `NarrowBar`, sides ϵb, b is the first of the thin regions treated by the Pack. In contrast to the usual account in engineering tables, the presence of the code for the rectangle makes it easy for the derivation to be presented, and, to lowest order, it is given in the Pack's documentation. Thus, the Pack gives the dominant term in S, which is that familiar to practising engineers. A research engineer, modifying the Pack's parameter specifying the number of terms to be taken in the series to ∞, and using the fact that for slender rectangles the tanh terms in the series can be replaced by 1, would soon find the improved approximation

$$S \sim \frac{16}{3}\epsilon^3 b^4 (1 - \frac{186\epsilon\zeta(5)}{\pi^5} + \frac{364\epsilon}{\pi^5}\exp(-\pi/\epsilon) + \dots \ . \qquad (11)$$

According to [Lo], p324, an approximation like this was known to St Venant last century.

(We remark that there are other places where one can use the fact that $\tanh(\lambda) \sim 1$ when λ is large to improve the numerical convergence of some series. For example, in the series for S for the rectangle, even when ϵ is not small, because of the presence of the summation index in the argument of the tanh, summing the series exactly with the tanh replaced by 1, and then taking into account the remainder might result in some small speed-up of the numerics. However, the cost in readability of code makes the decision not to use these devices in the Pack the correct one.)

For other thin regions, and for hollow shapes, see the Pack.

4 More Shapes for Exact Torsion Data?

We judge this Conference Proceedings as an inappropriate place to provide full printed details about exact solutions in more exotic domains. There is some unavoidable mess, and LaTeX may be the wrong publication medium, while CAS - via the Web - is a better medium. Some of the code will be made available via GK's Web pages.

In [KA] it was pointed out that most of the entries in the popular tables, e.g. [YR], were 'circular-arc polygons'. GK's project to collect exact solutions was described in [KA]. Collaboration with others across the Net is desired.

The remainder of this section, treating lens shapes, should be read with [KA].

4.1 Lens shapes, continued from AEMC96

The simplest case (other than the circle) of circular-arc polygons, involves just two arcs. A lens is an intersection of two disks. The semicircle given in the Basic Shapes in the Pack is a special case of a lens, so, once again, we see the Pack as a source of items against which to check developments.

4.1.1 Bipolar coordinates

An integral representation of the torsion function for a general lens is given in [LSU]. We have yet to evaluate the integrals in the general case. However for symmetric lenses, and u evaluated at the centre of the lens, CAS can be used to evaluate the integrals in certain special cases. We had earlier evaluated u_m for CC lenses (see [KA] and references there) and orthogonal lenses using different techniques. The results agree.

4.1.2 CC lenses

We now consider the CC family of lenses treated in [KA]. In the CC family, the centre of one disk is on the circumference of the other. (Normalise these so that the distance between the points of intersection of the two disks is 2.) One of the new results reported in [KA] was the torsional rigidity for the symmetric CC lens. In the notation of the [KA] paper the CC lens family is parametrised by a single parameter ξ_1 which runs between $\xi_1 = 0$ (a semicircle) up towards $\xi_1 = \pi$ (where the family is approaching a large disk). $\xi_1 = \pi/3$ is the symmetric CC lens and its torsional rigidity is reported in [KA]. The new integral evaluated in October 96 was $\xi_1 = \pi/2$. The exact results are summarised below.

Table 1: Torsional rigidities for some CC lens shapes

ξ_1		area	S_{CClens}
0	semicircle	$\pi/2$	$(\pi/8) - (1/\pi)$
$\pi/3$	symmetric	$(4\pi/9) - 2\sqrt{3}/3$	$(11\pi/36) - \sqrt{3}/2$
$\pi/2$		$\pi - 1$	$(161\pi/432) - 1$

4.1.3 Variational bounds

We have applied the method, which we tested with the Basic Shapes in §3.2.3, to lens shapes. The trial function (10) was used. The double integrals in (4) are treated as iterated integrals, and the y-integrals – integrating a quadratic in y – are done first. The formulae for the integrals for the subsequent x-integrals for general lenses – and even for the orthogonal and CC-lenses of [KA] – are messy (e.g. for orthogonal lenses in general they appear to involve elliptic integrals), and so the remaining x-integral is usually evaluated numerically. Huge simplifications occur for symmetric lenses, and the integrals $S_{LB,sym} = S_{LB}(v_{SymLens})$ evaluate in terms of elementary functions. The approximation $S_{LB,sym}$ has the correct leading asymptotic term both when the symmetric lenses are thin and when they are nearly circular.

As in the semicircle case reported in §3.2.3, we find that all these approximations are no more than 1% less than the true value of S. See also §4.3.

We leave open the problem of actually proving fairly general items related to this. We also speculate that, when such items are available, that they might be written clearly in some subsequent development of the Pack - or associated documents - in a way which engineering users find attractive, comprehensible, and easy to apply for shapes like the lens shapes of this subsection.

4.2 Perturbation expansions

Asymptotics for general near-circular shapes are treated in [PS]. See also [KA] for near-circular CC-lenses and §4.1.3 for symmetric lenses.

Asymptotics for thin and slender shapes from the Pack have been discussed in §3.3 and for symmetric lenses in §4.1.3.

Mathematica is useful here for the routine algebra.

4.3 Numerical approximation

This subsection reports progress on numerics in the direction anticipated in [KA] Section 5.

The Elastic Systems Pack provides general-purpose finite-element tools: see the Appendix for details.

Most engineers have access to numeric codes written in Fortran and in C, e.g. codes for solving the torsion problem, or, more generally, for solving the Dirichlet problem for Laplace's equation in the plane. GK had earlier used NAG's `D03EAF` routine to solve the torsion problem. `D03EAF` uses a boundary integral equation method and is ideal to return the torsional rigidity S using formulae given in [KA]. There are several routes to combine older codes like this with *Mathematica*.

A route we didn't use was using InterCall, see [HK]. InterCall provides a database and ready-to-use links to all routines in the NAG library. However, in contrast to [HK], the application in this paper is best treated with some further C code calling out to `D03EAF`: using InterCall would have required a preparation of an interface for this. Worse, it would have introduced another layer of third-party proprietary code, and we decided on a design which, while requiring C coding skills of people to adapt, would be flexible: thus we decided to use MathLink directly.

AT has written the code, following the design anticipated in [KA] §5. Thus MathLink calls to C which calls NAG `D03EAF`. The implementation is for lenses, with their vertices at (-1,0) and (1,0), and radii a and b. It would be easy to adapt the code to other circular arc polygons. The code is written to use the NAG library, but could be adapted to use some other torsion problem solver.

NAG's `D03EAF` had been used earlier (1989) for computations of S both for orthogonal lenses and for CC lenses. The 1996 computations repeated these and also treated symmetric lenses. The numerical results for symmetric lenses were compared with the lower bound of 4.1.3 and it is found that the simple explicit formula from the lower bound is about 1% less than the numerically computed values of the true S.

We speculate that symbolic-numeric combinations for computing torsional rigidity may be developed which are 'better' than, perhaps also faster than, the purely numeric codes like NAG's `D03EAF`. The increase in the number of *Mathematica*-using engineers, as a consequence of the various Packs, is bound to lead to experiments on this.

5 Qualitative PDE Work and CAS

One mathematical case for collecting examples of solutions of p.d.e. (partial differential equations) – the torsion problem, for example – in particular domains is that the mathematical literature abounds in guesses about how solutions behave for various classes of domains Ω – bounds on various combinations of functionals (§5.1), derivative behaviour (§5.2), etc..

The range of different items related to the torsion problem which arise in different applications is wide: see [KM, KA] for examples. Publication of exact torsion functions enables them to be reused in different applications.

5.1 Domain functionals

The older literature comparing S with other domain functionals – area $|\Omega|$, polar moments of inertia, etc. (which are available in another Chapter in the Pack) – is collected in [PS]. [PS] consider non-dimensional ratios R and how these vary over, say, convex Ω. In a very large number of cases, [PS] reports that certain of the Pack's Basic Shapes are extrema for R – slender sectors, equilateral triangles, ...

As an example of this general kind, here is a recent result, proved in [DK].

Theorem. *Let Ω be convex. Let z_c denote its centroid. Its torsion function u satisfies*

$$\frac{1}{9}u_m \leq u(z_c) \leq u_m \equiv \max_{\Omega} u,$$

$$\frac{1}{6} \le \frac{1}{9}\frac{u_m|\Omega|}{S} \le \frac{u(z_c)|\Omega|}{S} \le \frac{u_m|\Omega|}{S} \le 6.$$

To date, the extreme values we have found for $u(z_c)|\Omega|/S$ are 4/3 for a slender sector and 20/9 for an equilateral triangle. Our evidence for this came partly from pre-release code associated with the Pack.

5.2 Derivative behaviour

A case for exact solutions, as opposed to purely numerical solution, is when the behaviour of derivatives of the solution u rather than just u itself is central.

Consider the Hessian matrix D^2u. The concavity set Ω_1 is the set on which the Hessian matrix is negative semidefinite. The location of the maximum of the torsion function z_m is necessarily in Ω_1. The sector solution in the Pack can be used to show that, for sufficiently slender sectors, $z_c \notin \Omega_1$, and z_m is fairly close to the in-centre of Ω.

Manipulative mathematics occurs not just in connection with the 'classical' fare of CAS - exact solutions and perturbation expansions. For example, a small exercise which is easily done with a CAS is to show the following.

Theorem. *Let u be a solution of the torsion equation, and $H(u)$ the (determinant) of the Hessian of u. Then* $\log(1 - 4H(u))$ *is harmonic.*

There are uses of identities like this for example in connection with the concavity properties of solutions of the torsion problem for convex domains in the plane. See the survey paper of [KM] for details. Actually, one doesn't need a computer to get the particular result above, as one proof follows from the two-dimensional result that $\log(|\mathrm{grad}\phi|)$ is harmonic when ϕ is harmonic. However, there are lots of identities to check in this general area, see [KM], and it is easier to systematically ask the computer to use the same technique, eliminate y derivatives in favour of x derivatives using that u solves the torsion equation. It is just substitution, use of replacement rules. [KM] also gives references to where CAS have been used, in combination with Maximum Principles for elliptic p.d.e., to establish concavity results for other semilinear elliptic p.d.e..

Using CAS makes experimentation with messy identities less painful. Equally important is good human insight into what to try.

6 Conclusion, future prospects

The technology – Engineering Packs – is primarily for engineering use. However, applied mathematicians can find their own uses of them, and can also contribute to further development of the Packs.

Mathematicians are much involved with the basic algorithms of CA. The torsion problem is, of course, but one simple p.d.e. problem amongst many. The same technology - exact solutions, graphics, etc. - and so on are equally applicable for other problems in other Packs. As Engineering Packs acquire a large base of users, the funding and need may be there to build systematic mathematical libraries – e.g. a p.d.e. methods library ([LSU]) – to facilitate re-use of code in different Packs.

Many routes to further progress are opened by the production – and distribution – of the *Mathematica* Packs for engineers.

Acknowledgements

GK and AT thank Dan Moore and Imperial College London for the use of *Mathematica* 3, MathLink and NAG on Unix Workstations at Imperial College, London.

References

[Ce] C. Cetinkaya, "Engineering applications of computer algebra – Mathematica approach". In *Electronic Proceedings of the IMACS Conference on Applied Computer Algebra*, May 1995, University of New Mexico. `http://math.unm.edu/aca.html`

[DPU] W. Davis, H. Porta and J. Uhl, *"Calculus and Mathematica"* (Addison-Wesley: 1994).

[DK] S. Dragomir and G. Keady, "Generalisation of Hadamard's inequality for convex functions to higher dimensions, and an application to the elastic torsion problem". *University of Western Australia, Mathematics Research Report*, (1996).

[HK] E. Hawkes and G. Keady, "Inflation of a rubber disk: numerics using NAG d02raf". In *InterCall Manual and Examples Collection*, 1996. (Ed. P.Abbott and M.Trefry.)

[KA] G. Keady and P. Abbott, "Tables of stress and strain - the elastic torsion problem in a CAS". Pp 387-392 of *Proceedings of the Australian Engineering Mathematics Conference*, July 1996, Sydney. (Ed. P. Broadbridge.)

[KFW] G. Keady, N. Fowkes and J. Ward, "Interactive maths texts from Computer Algebra Systems: use in teaching engineering students", *Proc. of the Australian Computers in Education Conference*, July 1995, Perth, Vol 2, pp137-144. (Ed. R. Oliver and M. Wyld, ECAWA: 1995).

[KM] G. Keady and A. McNabb, "The elastic torsion problem in convex domains", *N.Z. Jnl of Mathematics* **22** (1993), 43-64.

[LSU] N.N. Lebedev, I.P. Skalskaya and Y.S. Uflyand, *"Problems of Mathematical Physics"* (Prentice-Hall: 1965).

[Lo] A.E.H. Love, *"A treatise on the mathematical theory of elasticity"* (4th ed., 1927: Dover reprinting, 1944).

[MH] J.H. Mathews and R.W. Howell, *"Complex analysis: for mathematics and engineering"* (Jones-Bartlett, 3rd ed.: 1997). (With Mathematica disk.)

[ON] P. O'Neil, *"Advanced Engineering Mathematics"* (PWS: 1996). (With Mathematica Lab Manual.)

[PS] G. Polya and G. Szego, *"Isoperimetric inequalities of mathematical physics"* (Princeton U. Press: 1951).

[So] I.S. Sokolnikoff, *"Mathematical theory of elasticity"* (McGraw-Hill, 2nd ed.: 1956).

[Vv] D.D. Vvedensky, *"Partial differential equations with Mathematica"* (Addison-Wesley: 1992).

[Wo] Wolfram, S. *"Mathematica: A system for doing mathematics by computer"* (Addison-Wesley, 3rd ed.: 1996).

[YR] W.C. Young and R.J. Roark, *"Roark's formulas for stress and strain"* (McGraw-Hill, 6th ed.: 1989).

[Zi] D.G. Zill, *"Differential Equations with Computer Lab Experiments"* (PWS, 1995). With accompanying PWS Notebook disks.

Appendix: General Description of the Elastic Systems Pack

by Cetin Cetinkaya

Elastic Systems is a collection of *Mathematica* packages addressing computational problems in analyzing elastic structural elements. In the design of the package, the symbolic capability of *Mathematica* is utilized so that the functions in the packages can be used as an interactive engineering "handbook." While many results are given in closed-form, some inherently numerical techniques, such as finite element method, are also incorporated.

The functions of Elastic Systems are designed for professional engineers, engineering educators, and engineering students of engineering mechanics in mechanical, civil, and aerospace engineering. In practice, the engineer can adopt the package to perform analysis of elastic structural elements along with other professional engineering analysis tools. These functions, coupled with the seamless computational features of the *Mathematica* environment, provide a powerful support tool for engineering analysis in many fields. Both *Mathematica* and the functions in this package are flexible problem-solving tools in which engineers and students examine both computational and mathematical issues involved in problem solving. Using the Elastic Systems functionality, many useful design methodologies can be implemented in the *Mathematica* environment.

The main engineering mechanics topics covered in the Elastic Systems package are as follows:

- Cross-sectional properties of two-dimensional shapes
- Bending of beams
- Torsional analysis of beams
- Two-dimensional finite element analysis
- Analysis of stress at a point
- Equations of elasticity theory

In engineering analysis and design, the cross-sectional properties of two-dimensional objects are widely used. The cross-sectional properties are essential, for example, in the bending of beams and torsional analysis. With this package, we can utilize both numerical and symbolic facilities of *Mathematica* in calculating the common cross-sectional attributes such as the area, the centroid, and moment of inertia of two-dimensional domain objects. These attributes for both basic domain objects and composite sections created by arranging the basic domain objects can be obtained. The basic domain objects include rectangular, triangular, circular, and elliptical sections, and parallelograms. Additional basic objects can be added to the package CrossSectionPropertiesSym if a need occurs. A composite section consists of a number of basic domain objects. The well-known composite cross sections such as T-sections, I-sections and channel sections are included in the package as built-in domain objects. For other composite cross sections such as L-sections and Z-sections, a simple procedure to create new cross sections from the basic domain is also included with a number of examples. The cross-sectional properties of these basic and composite sections can be calculated both symbolically and numerically. For more complex domains, the boundaries of a cross section are defined by using polygons. The same domain attributes can also be calculated for such complex domain, but only numerically, by the functions in the package CrossSectionPropertiesNum. The numerical techniques are based on a triangulation schema and an integration-in-a-triangle routine. Several examples of complex sections are also included.

In many engineering structures, cantilever beams are widely used as a main structural element receiving bending forces. The package BeamAnalysis provides a number of computational and graphical tools in stress analysis of cantilever beams. The package enables numerical and symbolic calculation of bending functions, stress fields, and deflection of a beam. The solutions included in BeamAnalysis are based on Saint-Venant's semi-inverse method in calculating stress functions. Since some of the formulas used in beam analysis are in series form, the symbolic results are useful examining the convergence of results, especially in stress computations. A number of two- and three-dimensional graphical functions can also be used to generate illustrative representations of deflected beams under bending loads.

Like bending of beams, torsional analysis has also been of practical interest to engineers for a long time. Structural elements subjected to twisting forces are used in many engineering applications. In the package TorsionAnalysis, closed-form solutions for a number of cross sections are available. The cross-sectional constants such as twist and torsional rigidity are calculated for these built-in cross sections. The stress and displacement fields as well as the torsion stress function can also be generated in closed-form. The package provides graphical tools to illustrate the visual aspects of the torsion analysis. The lines on a twisted bar, for example, can be shown using the function TorsionPlot. Also, the rotation of a cross section at an arbitrary station point can be drawn with respect to its original position. In addition to the TorsionAnalysis functions, many built-in *Mathematica* graphical functions are adopted to demonstrate various aspects of the package.

The main emphasis of the package is on the closed-form expressions, *given a shape*, for the torsion stress functions. We also present, via examples, a 'hybrid geometric-algebraic' method, used by Saint-Venant, to obtain stress functions for a class of cross sections. This method is based on assuming a form of harmonic function, e.g. polynomial, thence a solution of the torsion equation, and generating the shapes for the boundary conditions to be satisfied. The graphics from *Mathematica* make this more usable than formerly.

The finite element method has been adopted as the major analysis and design tool in many engineering fields. The modern commercial finite element programs offer many advanced computational features to solve a very large class of practical problems. In Elastic Systems, we introduce a set of tools that can be used to explore the main techniques of the finite element method such as constructing interpolation functions. In addition, a finite element code to calculate nodal forces and displacements for two-dimensional problems is included in the package along with a number of graphical tools to visualize the results. While this code is limited to plane stress and plane strain cases, material properties need not to be isotropic.

Using the functions, we can calculate one- and two-dimensional shape (interpolation) functions for Lagrange, Hermite, triangular, rectangular, and serendipity types of elements. Almost all shape functions found in textbooks can be generated with the help of these functions. Moreover, a general methodology for creating shape functions for new elements is also included. With this feature, users can design their own interpolation functions and elements, and test them against other elements using the functionality offered in this package. Such exercises in constructing user-defined interpolation functions can be invaluable to students of the finite element method as well as to practioners. Many graphical functions are provided for drawing explanatory pictures of the locations of nodal points and the forms of shape functions.

Solutions to a number of finite element problems from well-known textbooks are provided. The effect of meshing in finite element computations is discussed with many examples by solving the same problems with different meshes. As demonstrated in many examples, the mesh can also be generated with the built-in *Mathematica* programming constructs and mappings. These mappings are usually generated by interpolation functions. The results obtained in a finite element analysis session are illustrated with the aid of the graphical functions.

A number of tools to manipulate and to illustrate the state of stress at a point in a continuum are provided in the StressAnalysis package. The state of stress at a point in an elastic body is determined by six independent stress components. The stress state is specified by a second-order symmetric Cartesian tensor. This tensor is also known as the stress tensor. The values of these stress components change with the orientation of the axes system in which each stress component is defined. The axes system can be rotated such that some practical issues about the stress state at a point can be studied. For example, certain stress components can be reduced to zero in a particular orientation of the axes system.

Functions in the StressAnalysis package are provided to obtain useful information about the stress state at a point. For instance, the functions for computing the principal stress components and principal stress direction from a stress tensor are available. Also included are the functions for computing maximum shear stress and its directions and the graphical functions to plot the principal stress planes and principal stress directions. Mohr's circles can be drawn directly from the state tensor. A number of examples illustrating the use of these functions are provided.

The state of stress is usually studied to determine a single failure criterion from the stress tensor. From a number of proposed theories in predicting material failure, we consider three of the most commonly used:

- Maximum normal stress theory
- Maximum shear stress theory
- Distortion energy theory.

The equations of the linearized theory of elasticity, such as constitutive relation and equilibrium equations, are usually quite lengthy when they are expanded from their compact forms for an analysis. As a result, these equations are both difficult to remember and tedious to manipulate by hand except for overly simplified examples. The purpose of the package GoverningEquations presented in this chapter is to provide a systematic way to access these equations. The equations are formatted in two ways: text and indexed. In text form, the components are represented as *Mathematica* functions. For instance, `sigma[x,y]` represents the xy-stress component, σ_{xy}, in text form. In indexed form, the indices are used instead of the function brackets to specify the directions of stress, strain, and internal force components.

The package GoverningEquations provides equations in three different coordinate systems: Cartesian, polar, and spherical. There are also facilities to generate symbols for stress and stress components in a desired coordinate system. The functions for constitutive relation and equations for equilibrium and elastodynamics are included. A function for converting between various elastic coefficients is provided as well.