Kompleksilukujen kunnan konstruointi

Seuraava esitys osoittaa, miten kompleksilukujoukko voidaan määritellä tunnetuista reaalisista käsitteistä lähtien. Määrittelyjen jälkeen on helppoa osoittaa Mathematican laskentamahdollisuuksia käyttäen, että tulos on algebralliselta rakenteeltaan kunta.

Laskutoimistusten määrittely joukkoon $\mathbb{R}^2$

Lähtökohtana on xy-taso, ts. joukko $\mathbb{R}^2$, jonka alkioiden välille määritellään yhteen- ja kertolasku. Alkiot ovat lukupareja, Mathematican terminologialla kaksialkioisia listoja. Määrittelyä varten muodostetaan aluksi testi, joka tarkistaa, että funkion argumentit ovat sallittua tyyppiä:

```
In[1]:= ctest[z_] := Head[z] == List && Length[z] == 2
```

Laskutoimitukset ovat kahden argumentin funktioita, joiden arvona on argumenttien summa ja tulo siten kuin ne kompleksiluvuille määritellään:

```
In[2]:= summa[z_?ctest, w_?ctest] := {z[[1]] + w[[1]], z[[2]] + w[[2]]}

In[3]:= tulo[z_?ctest, w_?ctest] := {z[[1]] w[[1]] - z[[2]] w[[2]], z[[1]] w[[2]] + z[[2]] w[[1]]}
```

Merkintä $z_{[[k]]}$ viittaa listan z k:nteen alkioon.

Jotta päästään luonnollisempiin summan ja tulon merkintöihin, asetetaan Mathematicassa valmiiksi määritelty (mutta merkitystä vailla oleva) funktio `CirclePlus` tarkoittamaan funktiota `summa`, vastaavasti `CircleTimes` tarkoittamaan funktiota `tulo`. Tällöin laskutoimitusten symboleina voidaan käyttää renkaan sisässä olevia plus- ja kertomerkkejä (vinoristiä).

```
In[4]:= CirclePlus = summa;

In[5]:= CircleTimes = tulo;
```

Tämän jälkeen laskutoimitukset toimivat normaaliin tapaan:

```
In[6]:= {1, 2} ⊕ {3, 4}

Out[6]= {4, 6}

In[7]:= {1, 2} ⊗ {3, 4}

Out[7]= {-5, 10}
```

Nolla, ykkönen ja imaginaariyksikkö ovat erikoisasemassa olevia kompleksilukuja ja näille annetaan omat nimet:

```
In[8]:= o = {0, 0}

Out[8]= {0, 0}
```

In[9]:= `e = {1, 0}`

Out[9]= `{1, 0}`

In[10]:= `i = {0, 1}`

Out[10]= `{0, 1}`

Tälöin pätee

In[11]:= $i \otimes i == -e$

Out[11]= `True`

Tuloksena on saatu yhteen- ja kertolaskulla varustettu kompleksilukujen joukko $\mathbb{C}$.

Kuntastruktuuri

Joukko $\mathbb{C}$ voidaan osoittaa kunnaksi tarkistamalla kunta-aksioomien voimassaolo. Näissä esiintyy kolme mielivaltaisesti valittua kompleksilukua, joten luodaan kolme symbolista kaksialkioista listaa:

In[12]:= $a = \{\alpha_1, \alpha_2\}; b = \{\beta_1, \beta_2\}; c = \{\gamma_1, \gamma_2\};$

Näitä käyttäen tarkistetaan kunta-aksioomien voimassaolo yksi kerrallaan.

Yhteenlaskun vaihdannaisuus:

In[13]:= $a \oplus b == b \oplus a$

Out[13]= `True`

Yhteenlaskun liitännäisyys:

In[14]:= $(a \oplus b) \oplus c == a \oplus (b \oplus c)$

Out[14]= `True`

Yhteenlaskun neutraalialkio on nolla-alkio o:

In[15]:= $a \oplus o == a$

Out[15]= `True`

Alkion $a = \{\alpha_1, \alpha_2\}$ vasta-alkio löydetään ratkaisemalla yhtälö:

In[16]:= $x = \{\xi_1, \xi_2\};$

In[17]:= $\mathrm{Solve}[a \oplus x == o, x]$

Out[17]= $\{\{\xi_1 \to -\alpha_1, \xi_2 \to -\alpha_2\}\}$

In[18]:= `x /. %[[1]]`

Out[18]= $\{-\alpha_1, -\alpha_2\}$

Kertolaskun vaihdannaisuus:

In[19]:= `a⊗b == b⊗a`

Out[19]= `True`

Kertolaskun liitännäisyys, missä tarvitaan eksplisiittinen sievennyskäsky:

In[20]:= `(a⊗b)⊗c == a⊗(b⊗c)`

Out[20]= $\{(\alpha_1\beta_1-\alpha_2\beta_2)\gamma_1-(\alpha_2\beta_1+\alpha_1\beta_2)\gamma_2,\ (\alpha_2\beta_1+\alpha_1\beta_2)\gamma_1+(\alpha_1\beta_1-\alpha_2\beta_2)\gamma_2\} == \{-\alpha_2(\beta_2\gamma_1+\beta_1\gamma_2)+\alpha_1(\beta_1\gamma_1-\beta_2\gamma_2),\ \alpha_1(\beta_2\gamma_1+\beta_1\gamma_2)+\alpha_2(\beta_1\gamma_1-\beta_2\gamma_2)\}$

In[21]:= `% // Simplify`

Out[21]= `True`

Kertolaskun neutraalialkio on ykkösalkio `e`:

In[22]:= `a⊗e == a`

Out[22]= `True`

Alkion $a = \{\alpha_1, \alpha_2\}$ käänteisalkio löydetään ratkaisemalla yhtälö:

In[23]:= `Solve[a⊗x == e, x]`

Out[23]= $\left\{\left\{\xi_1\to\frac{\alpha_1}{\alpha_1^2+\alpha_2^2},\ \xi_2\to-\frac{\alpha_2}{\alpha_1^2+\alpha_2^2}\right\}\right\}$

In[24]:= `x /. %[[1]]`

Out[24]= $\left\{\frac{\alpha_1}{\alpha_1^2+\alpha_2^2},\ -\frac{\alpha_2}{\alpha_1^2+\alpha_2^2}\right\}$

Osittelulaki:

In[25]:= `a⊗(b⊕c) == (a⊗b)⊕(a⊗c)`

Out[25]= $\{\alpha_1(\beta_1+\gamma_1)-\alpha_2(\beta_2+\gamma_2),\ \alpha_2(\beta_1+\gamma_1)+\alpha_1(\beta_2+\gamma_2)\} == \{\alpha_1\beta_1-\alpha_2\beta_2+\alpha_1\gamma_1-\alpha_2\gamma_2,\ \alpha_2\beta_1+\alpha_1\beta_2+\alpha_2\gamma_1+\alpha_1\gamma_2\}$

In[26]:= `% // Simplify`

Out[26]= `True`

Koska kaikki kunta-aksioomat ovat voimassa, joukko $\mathbb{C}$ varustettuna edellä määritellyillä laskutoimituksilla on todellakin kunta.

Lisäyksiä laskutoimitusten määrittelyyn

Koska sekä summa että tulo ovat liitännäisiä, olisi luontevaa sallia sulkujen poisjättäminen useamman termin summassa ja useamman tekijän tulossa. Ilman lisämäärittelyjä tämä ei kuitenkaan onnistu, koska funktiot summa ja tulo on edellä määritelty vain kahden argumentin tapauksessa.

In[27]:= `a⊕b⊕c`

Out[27]= summa$[\{\alpha_1, \alpha_2\}, \{\beta_1, \beta_2\}, \{\gamma_1, \gamma_2\}]$

Tarvittavat lisämäärittelyt ovat seuraavat:

In[28]:= `summa[z_?ctest] := z`

In[29]:= `summa[z__?ctest, w__?ctest] := summa[summa[z], summa[w]]`

In[30]:= `tulo[z_?ctest] := z`

In[31]:= `tulo[z__?ctest, w__?ctest] := tulo[tulo[z], tulo[w]]`

Tämän jälkeen sulkujen poisjättäminen tai mielivaltainen ryhmittely toimii:

In[32]:= `a⊕b⊕c`

Out[32]= $\{\alpha_1 + \beta_1 + \gamma_1, \alpha_2 + \beta_2 + \gamma_2\}$

In[33]:= `a⊕b⊕c⊕a⊕b⊕c⊕a`

Out[33]= $\{3\,\alpha_1 + 2\,\beta_1 + 2\,\gamma_1, 3\,\alpha_2 + 2\,\beta_2 + 2\,\gamma_2\}$

In[34]:= `a⊗a⊗a`

Out[34]= $\{-2\,\alpha_1\,\alpha_2^2 + \alpha_1\,(\alpha_1^2 - \alpha_2^2), 2\,\alpha_1^2\,\alpha_2 + \alpha_2\,(\alpha_1^2 - \alpha_2^2)\}$

In[35]:= `lasku1 = (a⊗(a⊗a)⊗b)⊗(c⊗a)`

Out[35]=
$$\{-(\alpha_2\,((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2) + \alpha_1\,(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2))\,(\alpha_2\,\gamma_1+\alpha_1\,\gamma_2) +$$
$$(\alpha_1\,((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2) - \alpha_2\,(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2))\,(\alpha_1\,\gamma_1-\alpha_2\,\gamma_2),$$
$$(\alpha_1\,((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2) - \alpha_2\,(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2))\,(\alpha_2\,\gamma_1+\alpha_1\,\gamma_2) +$$
$$(\alpha_2\,((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2) + \alpha_1\,(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2))\,(\alpha_1\,\gamma_1-\alpha_2\,\gamma_2)\}$$

In[36]:= `lasku2 = a⊗(((a⊗a)⊗b)⊗c⊗a)`

Out[36]=
$$\{\alpha_1\,(-(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2)\,(\alpha_2\,\gamma_1+\alpha_1\,\gamma_2) +$$
$$((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2)\,(\alpha_1\,\gamma_1-\alpha_2\,\gamma_2)) -$$
$$\alpha_2\,(((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2)\,(\alpha_2\,\gamma_1+\alpha_1\,\gamma_2) +$$
$$(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2)\,(\alpha_1\,\gamma_1-\alpha_2\,\gamma_2)),$$
$$\alpha_2\,(-(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2)\,(\alpha_2\,\gamma_1+\alpha_1\,\gamma_2) +$$
$$((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2)\,(\alpha_1\,\gamma_1-\alpha_2\,\gamma_2)) +$$
$$\alpha_1\,(((\alpha_1^2-\alpha_2^2)\,\beta_1 - 2\,\alpha_1\,\alpha_2\,\beta_2)\,(\alpha_2\,\gamma_1+\alpha_1\,\gamma_2) +$$
$$(2\,\alpha_1\,\alpha_2\,\beta_1 + (\alpha_1^2-\alpha_2^2)\,\beta_2)\,(\alpha_1\,\gamma_1-\alpha_2\,\gamma_2))\}$$

In[37]:= `lasku1 - lasku2 // Simplify`

Out[37]= {0, 0}

Summan ja tulon laskujärjestystä ei tarvitse erikseen määritellä, sillä Mathematicassa on valmiina määrittely, jonka mukaan ⊗ lasketaan ensin ja ⊕ sen jälkeen:

In[38]:= `(a⊗b)⊕(a⊗c) == a⊗b⊕a⊗c`

Out[38]= True

Myös ei-negatiivinen kokonaislukupotenssi voidaan määritellä:

In[39]:= `pottest[n_] := IntegerQ[n] && NonNegative[n]`

In[40]:=
```
z_?ctest ⋄ n_?pottest := Nest[z ⊗ # &, e, n]
```

Huomaa, että symbolina ei ole tavallinen hattumerkki,vaan pieni vinoneliö!

In[41]:=
```
a ⋄ 2 // ExpandAll
```

Out[41]= $\{\alpha_1^2 - \alpha_2^2,\ 2\,\alpha_1\,\alpha_2\}$

In[42]:=
```
pot1 = a ⋄ 5 ⊗ b ⋄ 3 // ExpandAll
```

Out[42]= $\{\alpha_1^5\,\beta_1^3 - 10\,\alpha_1^3\,\alpha_2^2\,\beta_1^3 + 5\,\alpha_1\,\alpha_2^4\,\beta_1^3 - 15\,\alpha_1^4\,\alpha_2\,\beta_1^2\,\beta_2 + 30\,\alpha_1^2\,\alpha_2^3\,\beta_1^2\,\beta_2 - 3\,\alpha_2^5\,\beta_1^2\,\beta_2 - 3\,\alpha_1^5\,\beta_1\,\beta_2^2 + 30\,\alpha_1^3\,\alpha_2^2\,\beta_1\,\beta_2^2 - 15\,\alpha_1\,\alpha_2^4\,\beta_1\,\beta_2^2 + 5\,\alpha_1^4\,\alpha_2\,\beta_2^3 - 10\,\alpha_1^2\,\alpha_2^3\,\beta_2^3 + \alpha_2^5\,\beta_2^3,\ 5\,\alpha_1^4\,\alpha_2\,\beta_1^3 - 10\,\alpha_1^2\,\alpha_2^3\,\beta_1^3 + \alpha_2^5\,\beta_1^3 + 3\,\alpha_1^5\,\beta_1^2\,\beta_2 - 30\,\alpha_1^3\,\alpha_2^2\,\beta_1^2\,\beta_2 + 15\,\alpha_1\,\alpha_2^4\,\beta_1^2\,\beta_2 - 15\,\alpha_1^4\,\alpha_2\,\beta_1\,\beta_2^2 + 30\,\alpha_1^2\,\alpha_2^3\,\beta_1\,\beta_2^2 - 3\,\alpha_2^5\,\beta_1\,\beta_2^2 - \alpha_1^5\,\beta_2^3 + 10\,\alpha_1^3\,\alpha_2^2\,\beta_2^3 - 5\,\alpha_1\,\alpha_2^4\,\beta_2^3\}$

In[43]:=
```
pot2 = a ⊗ a ⊗ a ⊗ a ⊗ a ⊗ b ⊗ b ⊗ b // ExpandAll
```

Out[43]= $\{\alpha_1^5\,\beta_1^3 - 10\,\alpha_1^3\,\alpha_2^2\,\beta_1^3 + 5\,\alpha_1\,\alpha_2^4\,\beta_1^3 - 15\,\alpha_1^4\,\alpha_2\,\beta_1^2\,\beta_2 + 30\,\alpha_1^2\,\alpha_2^3\,\beta_1^2\,\beta_2 - 3\,\alpha_2^5\,\beta_1^2\,\beta_2 - 3\,\alpha_1^5\,\beta_1\,\beta_2^2 + 30\,\alpha_1^3\,\alpha_2^2\,\beta_1\,\beta_2^2 - 15\,\alpha_1\,\alpha_2^4\,\beta_1\,\beta_2^2 + 5\,\alpha_1^4\,\alpha_2\,\beta_2^3 - 10\,\alpha_1^2\,\alpha_2^3\,\beta_2^3 + \alpha_2^5\,\beta_2^3,\ 5\,\alpha_1^4\,\alpha_2\,\beta_1^3 - 10\,\alpha_1^2\,\alpha_2^3\,\beta_1^3 + \alpha_2^5\,\beta_1^3 + 3\,\alpha_1^5\,\beta_1^2\,\beta_2 - 30\,\alpha_1^3\,\alpha_2^2\,\beta_1^2\,\beta_2 + 15\,\alpha_1\,\alpha_2^4\,\beta_1^2\,\beta_2 - 15\,\alpha_1^4\,\alpha_2\,\beta_1\,\beta_2^2 + 30\,\alpha_1^2\,\alpha_2^3\,\beta_1\,\beta_2^2 - 3\,\alpha_2^5\,\beta_1\,\beta_2^2 - \alpha_1^5\,\beta_2^3 + 10\,\alpha_1^3\,\alpha_2^2\,\beta_2^3 - 5\,\alpha_1\,\alpha_2^4\,\beta_2^3\}$

In[44]:=
```
pot1 - pot2 // Simplify
```

Out[44]= {0, 0}

Siirtyminen kompleksilukujen tavanomaiseen esitykseen

Edellä on rakennettu kompleksilukuaritmetiikkaa Mathematicaan tarkoituksena osoittaa, miten tämä määritelmien pohjalta tulisi periaatteessa tehdä. Lopuksi on syytä katsoa, miten voidaan siirtyä kompleksilukujen tavanomaiseen esitykseen, mikä toki on Mathematicassa valmiinakin. Edellä käsitellyt kaksialkioisen listan muotoiset kompleksiluvut voidaan muuntaa normaaliin muotoon funktiolla

In[45]:=
```
normMuoto[z_?ctest] := z[[1]] + ⅈ z[[2]]
```

Tässä ⅈ tarkoittaa imaginaariyksikköä. Esimerkiksi:

In[46]:=
```
normMuoto[a]
```

Out[46]= $\alpha_1 + i\,\alpha_2$

In[47]:=
```
normMuoto[a ⋄ 2]
```

Out[47]= $\alpha_1^2 + 2\,i\,\alpha_1\,\alpha_2 - \alpha_2^2$

In[48]:=
```
ipot = Table[i ⋄ n, {n, 0, 10}]
```

Out[48]= {{1, 0}, {0, 1}, {-1, 0}, {0, -1}, {1, 0}, {0, 1}, {-1, 0}, {0, -1}, {1, 0}, {0, 1}, {-1, 0}}

In[49]:=
```
Map[normMuoto, ipot]
```

Out[49]= {1, ⅈ, -1, -ⅈ, 1, ⅈ, -1, -ⅈ, 1, ⅈ, -1}

Varsinaisesti kompleksiluvuilla laskettaessa ne syötetään Mathematicassa suoraan normaalimuodossa. Laskutoimitussymbolit ovat samoja kuin reaalisten muuttujien tapauksessa. Esimerkiksi:

```
In[50]:= (2 + 3 ⅈ) (4 + 5 ⅈ)
Out[50]= -7 + 22 ⅈ
```

```
In[51]:= (2 + 3 ⅈ) / (4 + 5 ⅈ)
```

Out[51]= $\frac{23}{41} + \frac{2\,i}{41}$

```
In[52]:= Table[ⅈ^n, {n, 0, 10}]
Out[52]= {1, ⅈ, -1, -ⅈ, 1, ⅈ, -1, -ⅈ, 1, ⅈ, -1}
```

Harjoitustehtäviä

1) Laske kompleksilukujen $z_1 = -5 + 7\,i$ ja $z_2 = -2 - 11\,i$ summa, tulo ja osamäärä sekä edellä olevan määritelmän mukaisella formalismilla että normaalia tapaa käyttäen.

2) Laske kompleksilukujen $z_1 = \cos(\varphi_1) + i\sin(\varphi_1)$ ja $z_2 = \cos(\varphi_2) + i\sin(\varphi_2)$ tulo ja osamäärä sekä luvun z_1 käänteisluku edellä olevan määritelmän mukaisesti. Sievennä tulokset.

3) Määrittele funktiot, jotka antavat reaalilukuparina esitetyn kompleksiluvun liittoluvun ja itseisarvon. Tarkista näitä käyttäen, että seuraavat yhtälöt ovat voimassa kaikille kompleksiluvuille: **a**) $\overline{z_1 + z_2} = \bar{z}_1 + \bar{z}_2$, **b**) $\overline{z_1\, z_2} = \bar{z}_1\, \bar{z}_2$, **c**) $\bar{\bar{z}} = z$, **d**) $|\bar{z}| = |z|$, **e**) $z\,\bar{z} = |z|^2$, **f**) $|z_1\, z_2| = |z_1|\,|z_2|$.

Simo K. Kivelä 23.4.2005