Quasigroups and Related Systems **14** (2006), 73 – 80

A short basis for the variety of WIP PACC-loops

J. D. Phillips

Abstract

The variety of weak inverse property, power-associative, conjugacy closed loops (WIP PACC-loops) is an ideal variety from which to investigate CC-loops. We give a surprisingly short basis for this variety. We also give a useful associator identity.

1. Introduction

A *quasigroup* $(Q,\cdot)$ is a set Q with a binary operation $\cdot$ such that for each a, $b \in Q$ the equations $a\cdot x = b$ and $y \cdot a = b$ have unique solutions x, $y \in Q$. A *loop* is a quasigroup with a two-sided neutral element, 1. We write xy instead of $x \cdot y$, and stipulate that $\cdot$ have lower priority than juxtaposition among factors to be multiplied—for instance, $x \cdot yz$ stands for $x(yz)$. For an overview of the theory of loops, see [3, 4, 17].

A loop is *conjugacy closed* (a *CC-loop*) iff it satisfies both of the following equations:

$$xy \cdot z = xz \cdot (z\backslash(yz)) \quad \text{(RCC)} \qquad\qquad z \cdot yx = ((zy)/z) \cdot zx \quad \text{(LCC)}$$

This definition is owing to Goodaire and Robinson [10, 11]; CC-loops were introduced independently, with different terminology, by Сойкис [20]. Further development of the theory can be found in [1, 2, 5, 6, 7, 14, 15]. The literature is not uniform as to which of these two equations is left (LCC) and which is right (RCC). With our choice here LCC is equivalent to requiring that the set of *left* multiplication maps be closed under conjugation. In [14, 15], the equation labels LCC and RCC were arranged in the opposite order.

The CC-loops which are also *diassociative* (that is every $\langle x, y\rangle$ is a group) are the *extra loops* introduced by Fenyves [8, 9]. For these, a detailed structure theory was described in [12], and their relationship to the other loops (and quasigroups) of so-called Bol-Moufang type is detailed in [18, 19].

2000 Mathematics Subject Classification: 20N05
Keywords: conjugacy closed loop, power-associative, weak inverse property

Definition 1.1. For any loop Q and for $c \in Q$:

1. Define c^ρ and c^λ by: $cc^\rho = c^\lambda c = 1$.
2. c is *power-associative* iff the subloop $\langle c \rangle$ is a group. Q is *power-associative* iff every element is power-associative.

The two parts of this definition are related by:

Lemma 1.2. (cf. [15]) *Let c be an element of a CC-loop Q. Then*

$$c \text{ is power associative} \quad \text{iff} \quad c^\rho = c^\lambda \quad \text{iff} \quad cc^2 = c^2 c$$

Power-associative CC-loops (PACC-loops) were thoroughly analyzed by Kinyon and Kunen in [13]. Central to their analysis is the notion of weak inverse property elements.

Definition 1.3. An element c of a loop Q is a *weak inverse property (WIP) element* iff $\forall x \in Q$,

$$c(xc)^\rho = x^\rho \qquad (cx)^\lambda c = x^\lambda . \tag{WIP}$$

A loop Q is then a *weak inverse property loop* if all of its elements have the weak inverse property.

Kinyon and Kunen showed that if Q is a PACC-loop and if $c \in Q$, then $c^{12} \in N(Q)$, the nucleus of Q; they prove this by showing that c^3 is a WIP element and c^6 is an extra element.

There is thus, a chain of five prominent varieties of CC-loops: (1) groups, (2) extra loops, (3) WIP PACC-loops, (4) PACC-loops, (5) CC-loops.

Obviously, a great deal has been written about the varieties (1), (2), and (5). And with the appearance of [13], the variety in (4) has now been analyzed in great detail; moreover, the prominence of the weak inverse property in the theory of CC-loops is now apparent. The variety in (3) is thus, a natural setting in which to investigate CC-loops: rich enough for deep structural theorems, but still, quite general. Unfortunately, the equations that axiomatize this variety are unwieldy. The main purpose of this paper is to give a surprisingly short and elegant basis for the variety of WIP PACC-loops.

2. Main result

Theorem 2.1. *In the variety of loops, WIP PACC-loops are axiomatized by the identities*

$$(xy \cdot x) \cdot xz = x \cdot ((yx \cdot x)z) \qquad \text{(LWPC)}$$

and

$$zx \cdot (x \cdot yx) = (z(x \cdot xy)) \cdot x. \qquad \text{(RWPC)}$$

Proof. We first show that WIP PACC-loops satisfy both LWPC and RWPC. We proceed in four steps. Firstly, we note that in WIP PACC loops, it is easy to see that both $(x/y)\cdot yz = y\cdot((y\backslash x)z)$ and $(x/y)\backslash 1 = y(x\backslash 1)$ hold, and that together they imply $x \cdot ((x\backslash y)z) = ((y\backslash 1)\backslash(x\backslash 1)) \cdot ac$. Secondly, we note that in WIP PACC loops, it is easy to see that $(x\backslash y)\ 1 = (y\backslash 1)x$, $((x\backslash y)\backslash 1)x \cdot (x\backslash y) = (x\backslash y)\backslash y$, and $x/y = (x\backslash 1)\backslash(y\backslash 1)$ hold, and that together they imply $(x\backslash y)\backslash y = (y\backslash 1) \cdot xy$. Thirdly, we note that in WIP PACC loops, it is easy to see that $(xy)\backslash 1 = y\backslash(x\backslash 1)$, $(x\backslash y)\backslash y = (y\backslash 1) \cdot xy$, and $(x\backslash y)z = ((y\backslash 1)\backslash(x\backslash 1) \cdot xz)$ hold, and that together they imply $(x\backslash 1)(xy \cdot x) = yx$. Fourthly, and finally, we note that in WIP PACC loops, it is easy to see that both $xy\backslash 1 = y\backslash(x\backslash 1)$ and $x\backslash(yx) = (x\backslash 1)y) \cdot x$ hold and that together the three identities just established, implies LWPC. Of course, dualize this proof to obtain RWPC.

We now show that loops satisfying both LWPC and RWPC are, in fact, WIP PACC loops. In LWPC, replace z with $x\backslash z$ to obtain $(xy \cdot x)z = x((yx\cdot x)\cdot(x\backslash z))$. Now, replace y with $(y/x)/x$ to obtain $(x((y/x)/x)\cdot x)z = x(y \cdot (x\backslash z))$. In other words, there exists a function $f(x, y)$ such that $L(x)^{-1}L(y)L(x) = L(f(x, y))$. Thus, in the variety of loops, LWPC implies LCC. Similarly, RWPC implies RCC. Also, it is easy to see that LWPC implies $x^\rho = x^\lambda$, which by Lemma 1.2 guarantees power-associativity. Finally, we offer an Otter output file proving that LWPC and RWPC together imply WIP:

```
Length of proof is 89.  Level of proof is 20.
---------------- PROOF ----------------
240 [] x=x.
242,241 [] 1*x=x.
244,243 [] x*1=x.
245 [] ((x*y)*x)*(x*z)=x*(((y*x)*x)*z).
246 [] (x*y)*(y*(z*y))=(x*(y*(y*z)))*y.
247 [copy,246,flip.1] (x*(y*(y*z)))*y=(x*y)*(y*(z*y)).
250,249 [] 1/x=x\1.
252,251 [] x*(x\y)=y.
254,253 [] x\(x*y)=y.
255 [] (x*y)/y=x.
```

```
258,257 [] (x/y)*y=x.
259 [] ((x*y)/x)*(x*z)=x*(y*z).
261 [] (x*y)*(y\(z*y))=(x*z)*y.
262 [] A*((B*A)\1)!=B\1.
263 [copy,245,flip.1] x*(((y*x)*x)*z)=((x*y)*x)*(x*z).
264 [copy,261,flip.1] (x*y)*z=(x*z)*(z\(y*z)).
265 [para_into,245.1.1.1.1,243.1.1,demod,242]
    (x*x)*(x*y)=x*((x*x)*y).
267 [para_into,245.1.1.2,243.1.1,demod,244]
    ((x*y)*x)*x=x*((y*x)*x).
271 [para_into,247.1.1.1,241.1.1,demod,242]
    (x*(x*y))*x=x*(x*(y*x)).
274,273 [para_into,251.1.1,241.1.1] 1\x=x.
285 [para_into,253.1.1.2,243.1.1] x\x=1.
287 [para_into,255.1.1.1,251.1.1] x/(y\x)=y.
298,297 [para_into,257.1.1.1,249.1.1] (x\1)*x=1.
302,301 [para_from,257.1.1,253.1.1.2] (x/y)\x=y.
305 [para_into,259.1.1.1.1,251.1.1] (x/y)*(y*z)=y*((y\x)*z).
309 [para_into,259.1.1.2,251.1.1] ((x*y)/x)*z=x*(y*(x\z)).
310 [copy,309,flip.1] x*(y*(x\z))=((x*y)/x)*z.
315,314 [para_into,261.1.1.2.2,257.1.1,flip.1]
    (x*(y/z))*z=(x*z)*(z\y).
316 [para_into,261.1.1.2.2,251.1.1]
    (x*(y\z))*((y\z)\z)=(x*y)*(y\z).
326 [para_into,263.1.1.2,251.1.1,flip.1]
    ((x*y)*x)*(x*(((y*x)*x)\z))=x*z.
343 [para_into,265.1.1.2,251.1.1,flip.1] x*((x*x)*(x\y))=(x*x)*y.
346,345 [para_into,265.1.1.2,243.1.1,demod,244] (x*x)*x=x*(x*x).
359 [para_from,267.1.1,255.1.1.1] (x*((y*x)*x))/x=(x*y)*x.
369 [para_into,271.1.1.1.2,251.1.1,flip.1]
    x*(x*((x\y)*x))=(x*y)*x.
409,408 [para_into,287.1.1,249.1.1] (x\1)\1=x.
423,422 [para_from,297.1.1,261.1.1.2.2,flip.1]
    (x*(y\1))*y=(x*y)*(y\1).
425 [para_from,297.1.1,264.1.1.1,demod,242,flip.1]
    ((x\1)*y)*(y\(x*y))=y.
428,427 [para_from,297.1.1,261.1.1.1,demod,242]
    x\(y*x)=((x\1)*y)*x.
430 [back_demod,425,demod,428] ((x\1)*y)*(((y\1)*x)*y)=y.
453 [para_from,314.1.1,267.1.1.1,demod,258]
    ((x*x)*(x\y))*x=x*(y*x).
512,511 [para_into,359.1.1.1.2.1,257.1.1,demod,315]
    (x*(y*x))/x=(x*x)*(x\y).
554,553 [para_into,422.1.1.1,297.1.1,demod,242,409,flip.1]
```

```
    (x*x)*(x\1)=x.
560,559 [para_from,427.1.1,251.1.1.2] x*(((x\1)*y)*x)=y*x.
565 [para_from,430.1.1,255.1.1.1] x/(((x\1)*y)*x)=(y\1)*x.
572,571 [para_into,453.1.1.1.2,253.1.1] ((x*x)*y)*x=x*((x*y)*x).
574,573 [para_into,453.1.1,422.1.1,demod,346,242]
    (x*(x*x))*(x\1)=x*x.
579 [para_from,553.1.1,427.1.1.2,demod,409,574] (x\1)\x=x*x.
599 [para_into,559.1.1.2.1.1,408.1.1] (x\1)*((x*y)*(x\1))=y*(x\1).
602,601 [para_into,559.1.1.2.1,559.1.1,demod,423,409,423,flip.1]
    ((x*y)*x)*(x\1)=x*((y*x)*(x\1)).
603 [para_into,559.1.1.2.1,251.1.1,flip.1] ((x\1)\y)*x=x*(y*x).
624,623 [para_into,579.1.1.1,408.1.1] x\(x\1)=(x\1)*(x\1).
636,635 [para_from,603.1.1,255.1.1.1,demod,512,flip.1]
    (x\1)\y=(x*x)*(x\y).
646,645 [para_into,305.1.1.1,249.1.1] (x\1)*(x*y)=x*((x\1)*y).
678 [para_into,635.1.1,253.1.1,flip.1] (x*x)*(x\((x\1)*y))=y.
680 [para_from,635.1.1,287.1.1.2] x/((y*y)*(y\x))=y\1.
684,683 [para_from,635.1.1,251.1.1.2] (x\1)*((x*x)*(x\y))=y.
686 [para_into,645.1.1.2,343.1.1,demod,684] (x\1)*((x*x)*y)=x*y.
692,691 [para_into,645.1.1.2,251.1.1,flip.1]
    x*((x\1)*(x\y))=(x\1)*y.
702,701 [para_from,645.1.1,253.1.1.2,demod,636,254]
    (x*x)*((x\1)*y)=x*y.
710,709 [para_into,309.1.1,243.1.1] (x*y)/x=x*(y*(x\1)).
720 [back_demod,310,demod,710] x*(y*(x\z))=(x*(y*(x\1)))*z.
725,724 [para_into,678.1.1.2.2.1,635.1.1,demod,554,636,254]
    ((x\1)*(x\1))*((x*x)*y)=y.
728 [para_from,678.1.1,645.1.1.2,flip.1]
    (x*x)*(((x*x)\1)*(x\((x\1)*y)))=((x*x)\1)*y.
733,732 [para_from,678.1.1,253.1.1.2] (x*x)\y=x\((x\1)*y).
734 [back_demod,728,demod,733,244,624,733,244,624]
    (x*x)*(((x\1)*(x\1))*(x\((x\1)*y)))=((x\1)*(x\1))*y.
740 [para_into,680.1.1.2.2,301.1.1,demod,315,258]
    x/(x*(y\x))=(x/y)\1.
754 [para_from,686.1.1,265.1.1.2,demod,725]
    ((x\1)*(x\1))*(x*y)=(x\1)*y.
756 [para_into,316.1.1.1,251.1.1] x*((y\x)\x)=(y*y)*(y\x).
766,765 [para_into,701.1.1.2,343.1.1,demod,636,725,252]
    (x*x)*(((x\1)*(x\1))*y)=y.
769,768 [back_demod,734,demod,766] x\((x\1)*y)=((x\1)*(x\1))*y.
771,770 [back_demod,732,demod,769] (x*x)\y=((x\1)*(x\1))*y.
784 [para_into,720.1.1.2.2,285.1.1,demod,244,flip.1]
    (x*(y*(x\1)))*x=x*y.
816 [para_into,740.1.1.2.2,301.1.1,flip.1] (x/(x/y))\1=x/(x*y).
```

```
828,827 [para_from,754.1.1,247.1.1.1,demod,646,423,298,242,646]
    (x*((x\1)*y))*x=x*((x\1)*(y*x)).
843,842 [para_from,756.1.1,253.1.1.2,flip.1]
    (x\y)\y=y\((x*x)*(x\y)).
858,857 [para_from,784.1.1,427.1.1.2,demod,254,646,828,423,flip.1]
    x*((x\1)*((y*x)*(x\1)))=y.
862 [para_from,784.1.1,253.1.1.2] (x*(y*(x\1)))\(x*y)=x.
867 [para_from,816.1.1,287.1.1.2,demod,250,flip.1]
    x/(x/y)=(x/(x*y))\1.
889 [para_into,862.1.1.2,251.1.1] (x*((x\y)*(x\1)))\y=x.
894,893 [para_from,867.1.1,301.1.1.1,demod,636,302,315,258,flip.1]
    x/y=x*((x*y)\x).
911 [back_demod,565,demod,894,560] x*((y*x)\x)=(y\1)*x.
925 [back_demod,257,demod,894] (x*((x*y)\x))*y=x.
935 [para_into,889.1.1.1.2.1,635.1.1,demod,843,554,274,572,252,646]
    (x*((x\1)*(y*x)))\y=x\1.
951 [para_into,925.1.1.1.2.1,251.1.1] (x*(y\x))*(x\y)=x.
959 [para_into,935.1.1.1.2.1,889.1.1,demod,692,244,692,244,646,692,
    244,843,554,274] (x*((x\1)*(y*(x\1))))\y=x.
995 [para_from,951.1.1,720.1.1.2,flip.1]
    (x*((x*(y\x))*(x\1)))*y=x*x.
1011 [para_from,369.1.1,599.1.1.2.1,demod,602,646,858,flip.1]
     (x*((x\y)*x))*(x\1)=y.
1025 [para_from,911.1.1,253.1.1.2,demod,428,flip.1]
     (x*y)\y=((y\1)*(x\1))*y.
1034,1033 [para_into,1011.1.1.1.2.1,253.1.1] (x*(y*x))*(x\1)=x*y.
1048 [para_into,1033.1.1.1,559.1.1] (x*y)*(y\1)=y*((y\1)*x).
1050 [para_into,1033.1.1.2,889.1.1,demod,692,244,692,244,692,244]
     (x\1)*(y*(x\1)))*x=(x\1)*y.
1068 [para_into,1048.1.1.2,770.1.1,demod,244,771,244,766]
     (x*(y*y))*((y\1)*(y\1))=x.
1080 [para_from,1050.1.1,326.1.1.2.2.1.1,demod,828,423,858]
     x*(y*((((y\1)*x)*y)\z))=y*z.
1092 [para_into,1080.1.1.2.2,1025.1.1,demod,560]
     x*((((y\1)*x)\1)*y)=y*y.
1098 [para_into,1092.1.1.2.1.1,251.1.1,demod,636]
     ((x*x)*(x\y))*((y\1)*x)=x*x.
1102 [para_into,1098.1.1.2.1,889.1.1,demod,692,244]
     ((x*x)*(x\(y\1)))*(y*x)=x*x.
1110 [para_from,1102.1.1,935.1.1.1.2.2]
     ((x*y)*(((x*y)\1)*(y*y)))\((y*y)*(y\(x\1)))=(x*y)\1.
1257 [para_into,1068.1.1.2.1,959.1.1,demod,242,692,244,242,692,244,
     242,692,244,843,554,274] (x*((y\1)*(y\1)))*(y*y)=x.
1263 [para_into,1257.1.1.1,995.1.1,demod,771,843,554,274,843,554,
```

```
     274,1034] (x*x)*(y*y)=x*(x*(y*y)).
1266,1265 [para_from,1263.1.1,765.1.1.2,demod,702]
     x*((x\1)*(y*y))=y*y.
1272,1271 [back_demod,1110,demod,1266,254,flip.1] (x*y)\1=y\(x\1).
1295 [back_demod,262,demod,1272,252] B\1!=B\1.
1296 [binary,1295.1,240.1] $F.
------------ end of proof -------------
```

Problem 2.2. *Is a quasigroup that satisfies* LWPC *and* RWPC *a loop?*

Recall, that an element c of a CC-loop is a *Moufang element* iff for all x, y $(c \cdot xy)c = cx \cdot yc$ (see [4],VII§2), and c is an *extra element* iff for all x, y $c(x \cdot yc) = (cx \cdot y)c$ [13]. Also, recall that the *associator* (x, y, z) is given by $xy \cdot z = (x \cdot yz) \cdot (x, y, z)$. Finally, given a loop L and an element $x \in L$, define the right and left multiplications by $xy = xR_y = yL_x$. We then may define the *inner mappings* on L by:

$$R(x, y) := R_x R_y R_{xy}^{-1}, \qquad L(x, y) := L_x L_y L_{yx}^{-1}, \qquad T_x := R_x L_x^{-1}.$$

Theorem 2.3. *Let c be a Moufang element in a CC-loop L. Then $\forall x, y$ we have $(c, x, y)^2 = (c^2, x, y)$.*

Proof. The otter output file of this proof may be found here:
`http://persweb.wabash.edu/facstaff/phillipj/research.html` □

Let c be an extra element in a CC-loop. Then $\forall x, y$ we have $(c, x, y)^2 = 1$ [13]. We now give a version of this result for Moufang elements.

Corollary 2.4. *Let c be a Moufang element in a WIP CC-loop. Then $\forall x, y$ we have $(c, x, y)^2 = 1$.*

Proof. In WIP CC loops, squares are nuclear [14]. Now, use Theorem 2.3. □

Acknowledgement. Our investigations were aided by the automated reasoning tool OTTER [16]. We thank Michael Kinyon for his helpful and generous comments.

References

[1] **А. С. Басараб**: *Об одном классе G-луп*, Математические Исследования Том **3**, Вып. 2 (8) (1968), 72 – 77.

[2] **А. С. Басараб**: *Класс* LK-*луп*, Математические Исследования, Вып. **120** (1991), 3 – 7.

[3] **В. Д. Белоусов**: Основы Теории Квазигрупп и Луп, Издательство «Наука», Москва, 1967.

[4] **R. H. Bruck**: A Survey of Binary Systems, Springer, 1971.

[5] **P. Csörgő and A. Drápal**: *Left conjugacy closed loops of nilpotency class* 2, preprint.

[6] **A. Drápal**: *Conjugacy closed loops and their multiplication groups*, J. Algebra **272** (2004), 838 – 850.

[7] **A. Drápal**: *Structural interactions of conjugacy closed loops*, preprint.

[8] **F. Fenyves**: *Extra loops I*, Publ. Math. Debrecen **15** (1968), 235 – 238.

[9] **F. Fenyves**: *Extra loops II*, Publ. Math. Debrecen **16** (1969), 187 – 192.

[10] **E. G. Goodaire and D. A. Robinson**: *A class of loops which are isomorphic to all loop isotopes*, Canadian J. Math. **34** (1982), 662 – 672.

[11] **E. G. Goodaire and D. A. Robinson**: *Some special conjugacy closed loops*, Canadian Math. Bull. **33** (1990) 73 – 78.

[12] **M. K. Kinyon and K. Kunen**: *The structure of extra loops*, Quasigroups and Related Systems **12** (2004), 39 – 60.

[13] **M. K. Kinyon and K. Kunen**: *Power-associative conjugacy closed loops*, J. Algebra, to appear.

[14] **M. K. Kinyon, K. Kunen, and J. D. Phillips**: *Diassociativity in conjugacy closed loops*, Comm. Algebra **32** (2004), 767 – 786.

[15] **K. Kunen**: *The structure of conjugacy closed loops*, Trans. Amer. Math. Soc. **352** (2000), 2889 – 2911.

[16] **W. W. McCune**: *OTTER 3.3 Reference Manual and Guide*, Argonne National Laboratory Technical Memorandum ANL/MCS-TM-263, 2003; `http://www.mcs.anl.gov/AR/otter/`

[17] **H. O. Pflugfelder**: Quasigroups and Loops: Introduction, Sigma Series in Pure Math. **8**, Heldermann, 1990.

[18] **J.D. Phillips and P. Vojtěchovský**: *The varieties of loops of Bol-Moufang type*, Algebra Universalis, to appear.

[19] **J.D. Phillips and P. Vojtěchovský**: *The varieties of quasigroups of Bol-Moufang type*, J. Algebra **293** (2005), 17 – 33.

[20] **Л. Р. Сойкис**: *О специальных лупах*, in Вопросы Теории Квазигрупп и Луп (В. Д. Белоусов, ed.), Редакц.-Издат. Отдел Акад. Наук Молдав. ССР, Кишинев, 1970, 122 – 131.

Received March 20, 2006

Department of Mathematics & Computer Science, Wabash College, Crawfordsville, IN 47933, U.S.A.
e-mail: phillipj@wabash.edu