A Nonvolatile Approximate Lookup Table Optimized for GPU Acceleration

Snehasis Dey,
Assistant Professor,
Dept.of Electronics & Telecommunication Engineering
College of Engineering Bhubaneswar

***Abstract*—In this letter, we design a nonvolatile approximate lookup table, named NVALT, to greatly accelerate general public utilities (GPUs) computation. Our architecture maintains high fre- quency input patterns within an approximation NVALT to simulate each application's functionality. To provide an approximative output, NVALT looks for and returns the stored data that most closely matches the input data. By taking use of the analog characteristics of the nonvolatile content addressable memory, we define a similarity measure that is suitable for binary representation. To regulate the precision required for user requirements, our approach manages the ratio of the application executing between the approximate NVALT and correct GPU cores. Our tests on seven common GPU applications demonstrate that, on average, NVALT may increase energy computation by 4.5 and performance by 5.7 while offering an average relative inaccuracy of less than 10% when compared to the baseline GPU.**

I. Introduction

INTERNETofThings(IoT)increasesthenumberofsmartdevices and the rate of data generation around the world [1],creatingasignificantdemandforhighspeed,andefficientparallelprocessors,suchasgeneralpublicutilities(GPU).However,GPUscannotlearnexistingpatternsinworkloadandadaptivelyprocesthedata[2].Whenrepeatedlyrun-ningasingleapplication,e.g.,fastFouriertransform(FFT),onaGPUfordifferentinputdata,theGPUperformsasetofcomputationsrepeatedlywithoutlearningthefunctionality.Byperformingnonconventionalbrain-likecomputation,e.g.,neuralnetworkandneuromorphiccomputingsolutionsper-form,thecoresavoidcostlyrepeatedcomputationsbylearningsetsoffunctionalityandapproximatelymodelthem[3].Thistechniqueiseffectiveforapplicationswhichcanacceptalevelofapproximation.Manyrealworldalgorithms,suchasmachinelearning,arestatisticinnatureandwillacceptsomeinaccuracyintheircomputation.Severalimageandvideoprocessingapplicationsaccepterrorincomputationwithoutlosingtheprecisionrequiredbyuser[2].Weaddaneffi cient brain-like processing unit beside the GPU cores to acceleratethecomputationwithapproximationtechniques.

Prior work tried to improve efficiency and accelerate theGPU/CPU computation by enabling approximation, such asvoltageover-scaling[4],precisionscaling[5],designingapproximatecircuits[6],[7],andmemoizationusingresis-tiveacceleration[8]–[12].Workin[3]acceleratesprogramsby utilizing a neural network placed beside the GPU cores toproduce approximate results. Although in CPU this techniqueprovides a large energy/performance improvement [14], theadvantage of neurally based approximation in GPU is minor.Several other designs use the high density and zero leakagepower of nonvolatile memories for memoization or enablingapproximationinGPUandCPUcores[8],[10],[13].Work in [8] enabled GPU approximation by using content address-able memories (CAMs) beside floating point units (FPUs) andapproximately retrieving data at run-time. This is a promisingtechnique to save the GPU energy, but it cannot improve theperformance, and also does not have a large impact on theoverall GPU energy consumption, considering data movementandothernon-FPUunits.

Incontrast,weproposeahardwareapproximationtechniquethat uses resistive content addressable memories to acceleratetheGPUcomputation.Weexploittheanalogcharacteristicsof the nonvolatile approximate lookup table, called NVALT,with nearest distance search capability. Our design learns andmodels the functionality of different applications by storinghigh frequency patterns for each application on an NVALT. Atrun-time, instead of processing data on the inefficient GPUcore, our design searches NVALTto find the data most simi-lar to the input operands. Our similarity metric considers theimpactofeachbitindextoprovidethebestmatch.Ourexperimental evaluation on advanced micro device (AMD) SouthernIsland GPU architecture, running seven different applications,showsthe enhanced GPU, as compared to the baseline GPU,can achieve 5.7energy and 4.5performance improvement,whileprovidinglessthan10%qualityloss.

II. ProposedNvalt

A. NVALT*ProgramAcceleration*

GPUworkloadsexhibitsignificantdatasimilarityandlocal-ity. This locality increases as we go through the IoT domainwithlargerdatasets.Severalbasicprograms,suchasFFTorimageprocessingapplications,consistofrepeatedbuild-ing blocks, where each has many addition and multiplicationoperations. Running these algorithms on the conventional pro-cessors (e.g., CPU or GPU) is slow and requires large energyconsumption. These applications can be approximately pro-cessedontheNVALTblock.Weexploitdatalocalitytomodel

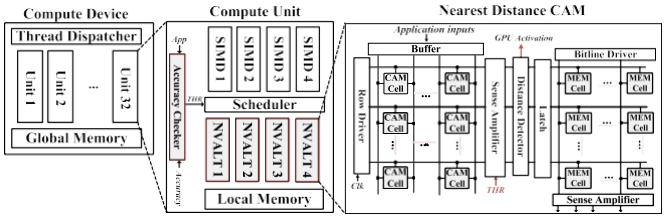


Fig.1. OverviewoftheenhancedGPUwithapproximateNVALTblock.

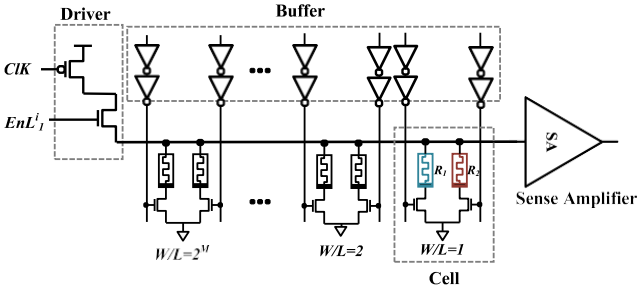
several basic functions on the GPU with a fast and efficientapproximatecomputingunit.

Fig.1showstheoverviewofthestructureofAMDSouthernIslandGPU,enhancedwithNVALTaccelerators.NVALT blocks are placed along side of each single instructionmultiple data (SIMD) lane. When the approximate applicationis launched on GPU, the scheduler block runs the applicationonanNVALTblockinsteadofsendingthemtoGPUcore sto process. An NVALTblock approximately processes thesebasic functions inside a lookup table instead of precisely pro-cessing them. NVALTuses offline profiling for each programto find and store common input patterns and the correspond-ing outputs. At runtime, our design searches the input datastoredintheNVALTandreturnsthemostsimilarpatternasan approximate output. The accuracy of depends two mainmetrics.

1) *Similarity Metric:* In traditional cores the numbers rep-resent by the fixed-point or floating point binary values,where the definition of similarity can be different basedon representation. For instance, our prior work [10] con-siders Hamming distance as a metric to find the mostsimilarrowinalookuptable.Thismetriccannotprovide highcomputationaccuracybecauseitdoesnotcon-sider the impact each bit index has on the computation.Designing a lookup table/CAM which can find the exactnearest value requires large and inefficient peripheralcircuitry. In this letter we propose a simple techniquewhich changes the weight of each bit index on the CAMstructure and considers the binary weights on the searchoperation.WedetailtheimplementationinSectionII-B.
2) NVALT *Size:* The lookup table size directly impacts thecomputation accuracy. A large size NVALT can store alarge number of patterns and increase the chance of aclose match. Larger lookup tables result in the increaseenergyanddegradedperformance.

NVALTworks for the applications which accept approxi-mation in their computation. Applications must have limitednumber of input/output signals to efficiently fit within theNVALT table. We model the computation of these applicationsby storing their highly frequent input/output patterns inside atable. Profiling mode examines of the input data to find themostcommonoccurrences.

Our evaluation shows that running all applications on theapproximate NVALT may not result in high energy saving orspeedup.WeneedtoincreasethesizeoftheNVALTtostorea sufficient number of patterns to ensure the acceptable com-putation accuracy. The increased size reduces the energy andperformance efficiency that NVALT achieves during the searchoperation. The low energy and performance advantage of neu-ral network-based GPU acceleration in [3] comes from usinglarge and inefficient neural network blocks to provide



enoughcomputation accuracy. To address this issue, this letter pro-videsconfigurableGPUapproximation,whereafractionof

Resistance value and select lines voltage during the search

Data	Conv. Cell		NVALT Cell		Search Operation	
	R1	*R2*	*R1*	*R2*	SL/SL'	ML
0	L	H	H	L	Vdd/0	Precharge
1	H	L	L	H	0/Vdd	

Fig. 2.Structure of CAM in NVALTblock capable of searching for nearestdistancerow.

datarunsontheaccurateGPUcore,whiletherestisrunontheapproxi mateNVALT.OurdesigncalculatesthedistanceoftheinputdatawithstoredvalueontheNVALT,andifthe distance is larger than a threshold (>THR), it dynamicallyassignsthisinputdatatoanaccuratecore.Thevalueof THRis determined based on the application type and user accuracyrequirement.

B. NVALT*HardwareSupport*

A conventional CAM consists of a CAM cell array, rowdriver, input buffer, and sense amplifier. Before each searchoperation,thematchlines(ML)ofallCAMrowsareprecharged to*Vdd*.Duringthesearchoperation,inputdatais distributed to all CAM rows using an input buffer. Thisbufferstrengthstheinputsignalstoensureallrowsreceivethe input signals at a similar time. During the search opera-tion, the MLs in all CAM rows are discharged as long as atleastonebitdifferenceexists.Weuseatimingcharacteristicof ML discharging current to differentiate rows with differentnumber of mismatches. In conventional CAM, any cell withmismatch discharges the ML. A larger number of mismatchesresults in a higher ML voltage drop, which can be detected bythe sense amplifier. To detect a row with minimum mismatch,i.e., closest Hamming distance row, we need to find the rowwhich discharges last. The sense amplifier tracks the ML volt-ages in all rows, until it find the slowest discharging one. Thedesign is complicated and requires additional circuity, such ascounters, while also taking a long time to determine the bestmatchedrow.

Toaddressthisissue,weuseinverseCAM,proposedin [10], to design a CAM with nearest distance search capa-bility. The CAM cells in our proposed NVALT work inversely,compare to the conventional CAM. Fig. 2 shows the function-ality of NVALT cells storing inverse resistance values in matchand mismatch cases. NVALT cell discharges the ML in case ofmatchingdatatothestoredvalues,whileamismatchMLstayscharged. A row which has more matched bits, i.e., a closestHamming distance row, creates a faster ML discharging cur-rent than other rows. We detect the nearest distance row byfindingthefirstrowtodischargetheML.

Fig.2showsaCAMinsingleNVALTstagewhichiscapableof searching for the row with the closest value to the providedinputs.TheproposedNVALTconsistsofaninputbuffer,a

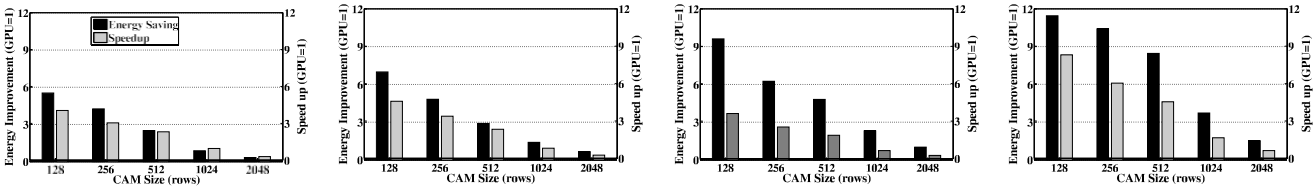


Fig.3.EnergyimprovementandspeedupoftheNVALTindifferentsizes.(a)BlackScholes.(b)FFT.(c)Sobel.(d)inversek2j.

row driver to selectively precharge ML of each row, and asense amplifier to detect the first discharged row. Unlike otherdesigns which use the hamming distance criteria, our designconsiders the impact of each bit index on the search operationoftheNVALTblock.Weexploitdifferentaccesstransistorsizesfor different bit indices. Based on the binary weight of anunsigned integer value, each cell in position *i*th has accesstransistorsxwhich are 2larger than the cell in the *i*1thadjacent bit. This resultx in 2higher ML discharging currentin each match cell compared to its adjacent least significantbits. The asymmetric access transistors weight each bit indexonthesearchoperationtofindtheclosetrow.

III. Experimental Results

A. *ExperimentalSetup*

We evaluate the efficiency of the proposed NVALT on theAMDSouthernIslandGPU,RadeonHD7970device,which is a recent GP-GPU architectures. We modified the sourcecode of Multi2sim, a cycle-accurate simulator [15], to inte-grate the NVALT within GPU. We used McPAT tool [16], tomeasuretheenergyconsumptionofGPGPUandotherchangesontheGPUarchitecture,includingtheregistersandfirst-input,first-ouput (4KB/SIMD). We also used CACTI to measure theenergy consumption of the memory keeping the highly frequencypatternscorrespondingtoeachapplication.Tomeasurethe power consumption of the NVALT, we use HSPICE circuitlevelsimulationin45nmTSMCtechnology.WeuseVTEAMmemristor model [17] for our memory design simulation withR_{ON}andR_{OFF}of10k▲and10M▲,respectively.Intermsof reliability, the endurance of resistive memory limits to 10^7writeoperations,ourdesignaddressesthisissuebylimitingthenumberofwritestoonceforeachapplication.Wecomparethe efficiency of proposed design by running seven generalOpenCL applications: *BlackScholes*, *FFT*, *Sobel*, *inversek2j*,*Laplacian*,*Convolution*,and*Binarization*.

B. NVALT*AccuracyTuning*

NVALTcan be used next to GPU SIMD lanes to approximately process different applications. We use the approximatebuilding block as a stand-alone computing unit to approx-imately process the desired applications. In this mode, theNVALT accuracy tunes using the size of the table that we areusing.AsweexplainedinSectionII,thereisanefficiencyandaccuracy tradeoff in choosing the best NVALT size to process the data. Small size table can provide significantly higherenergy savings and performance improvement at the cost ofdecreasedaccuracy.Increasingthetablesizeincreasesthecomputation accuracy by storing more common patterns, thusincreasing the possibility of close hit. However, NVALT withmanyrowsrequireslargerinterconnectsandcolumndriverstodistributethedatasimultaneouslyamongallCAMrows.TABLEI

TUNING ACCURACY IN ROW-TUNABLE NVALT BLOCK

#ofrows	128	256	512	1024	2048
BlackScholes	21%	13.1%	7.2%	4.0%	1.2%
FFT	8.3%	5.9%	3.1%	1.9%	0.7%
Sobel	13.4%	6.4%	4.2%	2.1%	1.4%
Inversek2j	17.3%	12.0%	8.3%	5.1%	2.9%

Thisenergyovertakesthesearchenergy(theenergyconsumedbythearray),forCAMlargerthan1024rows.

Our design is able to dynamically change the number ofactiverowsontheNVALT(128-rowgranularity)inordertoget the maximum accuracy for each application. We enable thisfunctionality by chaining the structure of the row driver. Fig. 3shows the energy and performance improvement of four generalapplicationsonthepropose denhancedhardware.Foreach CAM size, the energy and performance have been normalizedto the conventional GPU not using NVALT. Table I shows thecomputation accuracy of each application for different CAMsizes.Ourresultshowstheapplicationshavedifferentaccuracysensitivity to the NVALTsize, as each application requires adifferent number of rows to provide acceptable quality of ser-vice. To achieve less than 10% quality loss, our design needsto use an NVALTup to 512 rows. The result shows that ourdesign achieves 5.3energy improvement and 3.2speedup,when selecting the optimal NVALT size for each application.Themainissueoftherowtuningtechniqueisitsweaknessin providing energy/performance advantage for high qualitycomputation. To provide less than 4% quality loss, our designdoesnotsaveenergyorperformance.

C. *GPU*-NVALT*Computing*

To address the quality issue in row tuning technique, weneedtounderstandwhyverylargeNVALTblocksstillresult in sizable error. NVALT shows high error running inputs

thathave low similarity to the stored values. Although the per-centage of data with low similarity makes up a small fractionof each workload, the impact of them on accuracy is high. Inmost cases, these infrequent patterns of low similarity data arethe main information of the input. For example, in image pro-cessing, the image edges have the low frequency compared tothe image background, but the edges contain the main infor-mation of image. To improve the NVALT accuracy, our designdynamically finds the far input values and assigning them toaccurate SIMD cores to process. The ratio of running data onthe approximate and precise core determines the level of accu-racy that our design can achieve. Fig. 4 shows the maximumenergy improvement and speedup that our design can achievein each NVALTsize when the threshold value changes. TheTHRhasbeensetinordertoensurethatGPUqualitylossis

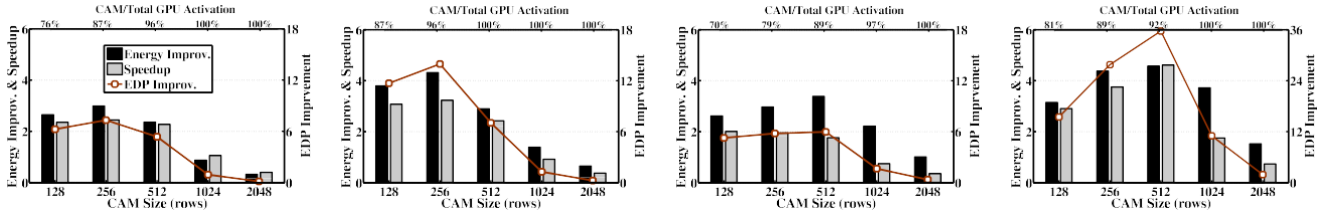


Fig.4. Energyimprovement, speedup, andEDP ofenhanced-GPU with NVALTindifferent CAMsizes while ensuringless than4% qualityloss. (a) BlackScholes.(b)FFT.(c)Sobel.(d)inversek2j.

TABLEII
EDPIMPROVEMENT OF THE ENHANCED-GPUWITH NVALT IN DIFFERENT QUALITY LOSS

Quality loss	1%	2%	4%	6%	8%	10%
Best CAM size	024-row	512-row	256-row	256-row	128-row	128-row
BlackScholes	1.1×	3.2×	7.3×	10.2×	12.3×	14.7×
FFT	2.7×	6.3×	13.9×	20.5×	23.9×	30.1×
Sobel	2.9×	9.9×	27.7×	38.6×	45.5×	54.0×
inversek2j	1.0×	2.8×	5.7×	8.4×	10.3×	13.3×
Laplacian	3.1×	5.2×	8.4×	12.3×	14.9×	19.3×
Convolution	8.6×	21.2×	32.8×	45.8×	61.4×	79.3×
Binarization	2.9×	5.8×	9.3×	13.5×	17.0×	22.1×
Average	**3.2**×	**7.6**×	**15.0**×	**21.3**×	**26.5**×	**33.2**×

lessthan4%.Thetopy-axisinthefigureshowsthepercentageoftimetheapplicationrunson CAMforagivenTHRvalue.Ourevaluationshowsthat,insmallNV ALTsize,weneedtorunlargerportionofdataonthepreciseGPUcore inordertoguaranteethecomputationaccuracysatisfiestherequired level.Bycontrast,inlargesize,theNVALTcanprovideenoughcomp utationaccuracyevenusingverylargethreshedvalue.Weconsider energy-delayproduct(EDP)tofindtheopti-malCAMsizeresultinginthebestefficiency.Ourevaluationshows that 256-row CAM results in the best efficiency of × 4.4energyimprovementand3.4× speedupacrosstheseventestedap plications(normalizedtoconventionalGPU).

Inallapplications,4%qualitylossdoesnotshowtheaccept-able quality of service. Therefore, we change the THR valuefor each application to see the efficiency of the NVALTindifferent level of accuracy. Table II shows the EDP improve-ment (normalized to GPU) that our design can achieve fordifferent quality loss from 2% to 10%, as the NVALTsizechanges.Theresultshowsthatourdesigncantunethele velofapproximationbypartiallyrunningtheapplicationonNVALT and accurate GPU core. Our result shows that NVALTworksmuchmoreefficientlyforapplicationswhichcanacc eptlarge approximation. For instance, by accepting 6% error, theNVALTcan achieve up to 3.7× speedup, while this numberdecreasesto1.4 × toachieve1%qualityloss.

IV. CONCLUSION

This letter suggests a method of hardware approximation that speeds up GPU architecture programs. In order to search for the closest distance row, we develop a (NVALT). NVALT stores high frequency patterns associated with each application, so roughly modeling computing. The stored value that bears the greatest resemblance to the input data is found by our NVALT algorithm. Through the dynamic division of input data between the correct GPU cores and the approximate NVALT, our approach is able to adjust the degree of approximation. NVALT can enhance the GPU, according to our experimental study that runs seven general applications on AMD GPUs.

Compared to the unmodified GPU, energy computation was reduced by 4.5 and performance by 5.7 on average, with less than 10% average relative error.

REFERENCES

[1] J.GantzandD.Reinsel,"Extractingvaluefromchaos,"*IDCiview*,vol.1142,p p.1–12,2011.

[2] M. Imani, Y. Kim, A. Rahimi, and T. Rosing, "ACAM: Approximatecomputing based on adaptive associative memory with online learn-ing,"in*Proc.IEEE/ACMISLPED*,SanFrancisco,CA,USA,2016,pp.162–167.

[3] A.Yazdanbakhsh, J. Park, H. Sharma, P. Lotfi-Kamran, and H. Esmaeilzadeh, "Neural acceleration for GPU throughput processors,"in*Proc.IEEE/ACMMicro*,2015,pp.482–493.

[4] L. Leem, H. Cho, J. Bau, Q. A. Jacobson, and S. Mitra, "ERSA: Errorresilientsystemarchitectureforprobabilisticapplications,"in*Proc.IEE /ACMDATE*,Dresden,Germany,2011,pp.1560–1565.

[5] S.Venkataramani,V.K.Chippa,S.T.Chakradhar,K.Roy,and A. Raghunathan, "Quality programmable vector processors for approx-imate computing," in *Proc. IEEE/ACM Micro*, Davis, CA, USA, 2013,pp.1–12.

[6] K. Nepal, Y. Li, R. I. Bahar, and S. Reda, "ABACUS: A technique forautomated behavioral synthesis of approximate computing circuits," in*Proc.IEEE/ACMDATE*,Dresden,Germany,2014,pp.1–6.

[7] M. Imani, D. Peroni, and T. Rosing, "CFPU: Configurable floating pointmultiplier for energy-efficient computing," in *Proc. IEEE/ACM DAC*,Austin,TX,USA,2017,Art.no.76.

[8] M. Imani, A. Rahimi, and T. S. Rosing, "Resistive configurable associa-tive memory for approximate computing," in *Proc. IEEE/ACM*

DATE,Dresden,Germany,2016,pp.1327–1332.

[9] X. Yin, M. Niemier, and X. S. Hu, "Design and benchmarking of ferroelectric FET based TCAM," in *Proc. IEEE/ACM DATE*, Lausanne,Switzerland,2017,pp.1444–1449.

[10] M. Imani, D. Peroni, A. Rahimi, and T. Rosing, "Resistive CAM acceleration fortunableapproximatecomputing,"*IEEETrans.Emerg.TopicsComput.*,tobepublished,doi:10.1109/TETC.2016.2642057.

[11] V.Akhlaghi,A.Rahimi,andR.K.Gupta,"Resistivebloomfil-ters: From approximate membership to approximate computing withbounded errors," in *Proc. IEEE/ACM DATE*, Dresden, Germany, 2016,pp.1441–1444.

[12] M.Imani,S.Patil,andT.S.Rosing,"MASC:Ultra-lowenergymultiple-accesssingle-chargeTCAM for approximate com-puting,"in*Proc. IEEE/ACM DATE*, Dresden, Germany, 2016,pp.373–378.

[13] M.Imani,S.Patil,andT.Rosing,"Approximatecomputingusingmultiple-accesssingle-chargeassociativememory,"*IEEETrans.Emerg.TopicsComput.*,tobepublished,doi:10.1109/TETC.2016.2565262.

[14] H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger, "Neural acceleration forgeneral-purposeapproximateprograms,"in*Proc.IEEE/ACMMicro*,Vancouver,BC, Canada,2012,pp.449–460.

[15] R. Ubal, B. Jang, P. Mistry, D. Schaa, and D. Kaeli, "Multi2Sim: A simulation framework for CPU-GPU computing," in *Proc. ACM PACT*,Minneapolis,MN,USA,2012,pp.335–344.

[16] S.Li*etal.*,"McPAT:Anintegratedpower,area,andtimingmod-eling framework for multicore and manycore architectures," in *Proc.IEEE/ACMMicro*,NewYork,NY,USA,2009,pp.469–480.

[17] S.Kvatinsky,M.Ramadan,E.G.Friedman,andA.Kolodny,"VTEAM:A generalmodelforvoltage-controlledmemristors,"*IEEETrans.CircuitsSyst.II,Exp.Briefs*,vol.62,no.8 ,pp.786–790,Aug.2015.