

ESTIMATING TAU (τ) FROM LEFT VENTRICULAR PRESSURE WAVEFORMS DURING
VENA CAVAL OCCLUSIONS.

A Paper
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Sushma Gopinath

In Partial Fulfillment
for the Degree of
MASTER OF SCIENCE

Major Department:
Electrical and Computer Engineering

December 2011

Fargo, North Dakota

North Dakota State University
Graduate School

Title

Estimating Tau (τ) from Left Ventricular Pressure Waveforms during Vena Caval Occlusions.

By

Sushma Gopinath

The Supervisory Committee certifies that this *disquisition* complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Dan Ewert

Chair

Dr. Mark Schroeder

Dr. Larry Mulligan

Dr. Sumathy Krishnan

Approved by Department Chair:

11 January 2012

Date

Dr. Rajendra Katti

Signature

ABSTRACT

Many people are dependent on artificial pacemakers to have a normal cardiac function. Due to this it is important to study the effects of pacing on cardiac function, as well as, to determine the best site to pace with an artificial pacemaker that yields the best cardiac performance. Tau (τ), the time constant of left ventricular relaxation, has been studied as one measure of effective cardiac function, where high τ has been associated with myocardial ischemia and hence a low τ is desirable. The objective of the current study was to create a program in MATLAB™ that estimates τ from left ventricular pressure (LVP) data, verify this program using synthesized data and calculate τ for physiological data. LVP data was collected from five canines under four pacing modes: left ventricular (LV), bi-ventricular (BV), right atrial (RA) and right ventricular (RV) at rates of 90 or 100 and 160 bpm. Four models of τ were used: 1. A semi-logarithmic, zero asymptote model (τ_L), 2. A semi-logarithmic model using data from the first 40ms of the isovolumic relation (τ_{40}), 3. Exponential model with non-zero asymptote of left ventricular pressure (τ_E) and 4. A derivative model with non-zero asymptote of left ventricular pressure (τ_D). The program successfully loaded all data files and computed τ for all dogs, all pacing sites and all heart rates.

ACKNOWLEDGMENTS

I would like to thank my Advisor, Dr. Ewert, for his support and guidance which made this paper possible. I would like to thank Dr. Mulligan and Medtronic for the data, without which this project could not have been done. I would also like to thank my committee members Dr. Schroeder and Dr. Krishnan and Graduate coordinator Dr. Kavasseri for their support and all my professors who have taught me during my Masters.

DEDICATION

This paper is dedicated to my parents K. Gopinath and Sarala Gopinath, without whose support, this paper would not have been possible.

TABLE OF CONTENTS

ABSTRACT.....	iii
ACKNOWLEDGMENTS.....	iv
DEDICATION.....	v
LIST OF TABLES.....	vii
LIST OF FIGURES.....	viii
INTRODUCTION.....	1
METHODS.....	3
RESULTS AND DISCUSSIONS.....	8
CONCLUSION.....	17
REFERENCES.....	18
APPENDIX I.....	20

LIST OF TABLES

<u>Table</u>	<u>Page</u>
1. Validation of the algorithm used.....	11

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
1. Flow-diagram of the algorithm of the program used.....	8
2. Sample waveform of unfiltered LVP.....	9
3. Plot of LVP Vs filtered and offset LVP.....	10
4. The LVP waveform of Dog 2 at VCO at pacing site R.A. with heart rate of 160 bpm with EDP and negative peak dP/dt marked.....	11
5. Plot of synthesized data.....	12
6. A single heartbeat with portions used for analysis of different methods	13
7. Comparison of τ calculated by four methods for Dog 1 at pacing site BV at a rate of 90 BPM during VCO.....	14
8. Curve fits for the LVP analyzed, using the four methods.....	15

INTRODUCTION

Many people suffer from various cardiovascular diseases and sometimes it results in alterations of the natural pace maker functions of the heart. When this happens, artificial pacemakers are used to pace the heart at sites like the left ventricle or bi- or right atrial or right ventricle. Due to this, it is important to study the effects of pacing on cardiac function to determine the best pacing site that yields better cardiac function. Different physiological mechanisms have been investigated to determine the state of the heart under various conditions. It was observed that prolongation of myocardial relaxation was an early sign of acute myocardial ischemia. [1] [2] This could be used as one diagnostic measure to determine the effects that various pacing sites have on the cardiac functionality. This rate of myocardial relaxation has been described by different mathematical models. One of them is the time constant of relaxation, Tau. Tau (τ) was first described by Frederiksen et al. as a time constant during the exponential fall of the isovolumic left ventricular pressure, after the peak negative pressure differentiated with respect to time [3]. τ was derived by Weiss et al. as the negative inverse of the slope of the plot of the natural logarithm of left ventricular pressure against time [4]. Rousseau et al. came up with a new model where they divided the isovolumic relaxation period into two 40 sec segments and treated each segment as an exponential function. They observed that the impaired isovolumic relaxation was mainly during the first 40 msec after peak (negative) dP/dt . Hence the model of τ proposed by them analyses the isovolumic relaxation in the first 40 ms after peak negative dP/dt as an exponential function similar to that proposed by Weiss et al. [5] These two models, however, did not take in to account the pressure changes from changing pericardial or pleural pressures. They assumed that the left ventricular pressure in the ventricular cavity would simply decline to zero asymptotically. To take in to account non-zero asymptote decline of the left

ventricular pressure, Raff et al. proposed a model which relates the left ventricular pressure to the first derivative of pressure with respect to time (dP/dt). In this model τ is derived by taking the negative inverse of the slope of the regression line of left ventricular pressure against left ventricular dP/dt . [6] Thompson et al. also proposed a non-asymptote model of τ . [7] They used an exponential method to calculate τ by considering three points equally spaced in time at 20 ms intervals during the fall of the left ventricular pressure between the peak negative dP/dt and the pressure corresponding to the end-diastolic pressure of the previous beat and then iteratively calculating τ . These models have been used by others to assess the different states of heart. The model proposed by Weiss et al was used to determine the myocardial stiffness during pacing induced angina. [8] All the other models of τ were used to evaluate the left ventricular performance during transluminal angioplasty. [9] These models were also used to determine the load independence of rate of isovolumic relaxation. [10] Though τ has been used to evaluate various mechanisms of the heart, not much work has been done to use T as a measure in determining the effects of artificial pacing through various sites. [11] [12]

The objective of the current study was to create a program in MATLAB™ that estimates τ from left ventricular pressure (LVP) waveforms, verify this program using synthesized data and calculate τ for LVP obtained from five dogs, paced at four pacing sites during vena caval occlusion by using the four methods discussed above to calculate τ .

METHODS

SURGICAL PROCEDURE:

This study was approved by the Medtronic Institutional Animal Care and Use Committee (Study S1288). The surgical procedure, instrumentation methods and experimental protocol have already been documented elsewhere [13], but will be briefly reiterated here for the benefit of this study. Five dogs, weighing at least 24 kgs and of both sexes were used in this study. 30 min before surgery, they were given an antibiotic, an analgesic and a sedative in the form of Cefazolin (700mg IV), morphine (1mg/kg IM) and Propofol (120 mg IV) respectively. 2% isoflurane was used to maintain anesthesia followed by an injection of succinylcholine (20mg IV). Then, under sterile conditions, a left thoracotomy was performed through either the fourth or fifth intercostal space to place an epicardial lead on the left ventricular free wall. Through a left jugular venotomy under fluoroscopic guidance, a lead was placed in the right ventricular apex and in the right atrial appendage. The chest was then closed and the leads were connected to a biventricular implantable pulse generator (Model 8042 Insync III, Medtronic, USA).

INSTRUMENTATION:

The animals were given a period of at least 2 weeks to recover. In preparation for measurement of pressure-volume (PV) data, they were given an antibiotic, Cefazolin(700 mg IV). During positioning of the instrumentation, Isoflurane (1.5-2%) was used to maintain the animal at an appropriate plane of anesthesia. A conductance catheter (CD Leycom, Netherlands) was introduced in to the left ventricle through a right femoral arteriotomy. The catheter (7 Fr, CA-72103-PNA), which operated in dual-frequency mode, was connected to a Sigma-5 DF (CD Leycom, Netherlands) control box to obtain the analog volume signal. The analog pressure output was acquired by connecting the pressure sensor to a Sentron Pressure Interface (Model

SPI-110, CD Leycom, Netherlands). A Fogarty occlusion catheter (62080814F, Edwards Lifesciences, USA) was introduced into the vena cava via a right femoral venotomy. The PV data was collected with fenantyl (5 µg/kg IV) infused at a rate of 2-5 ml/h with a CRI drug pump and isoflurane maintained at a low level (1%). While pacing atrium at a rate of 120bpm, the conductance-derived ventricular volume was calibrated by scaling and shifting the conductance signal to match ventricular volumes estimated by triplane echocardiography. This rate was selected for calibration because it was faster than the natural heart rate in all five dogs, and was also between the upper and lower limits of pacing rates used in this study. The monitoring of heart rate and capture verification was done through standard ECG limb leads. Black et al stated that “Left Ventricular pressure(LVP), volume(LVV), Marker ChannelTM, ECG limb leads one, two and three were digitized at 1,000 Hz with 12 bit analog to digital resolution using IOX software version 1.8.11(EMKA Technologies, France). Data was stored on a computer and then exported to text files for analysis.” [13]

EXPERIMENTAL PROTOCOL:

The canine hearts were paced at rates of 90 or 100 and 160 beats per minute using a Medtronic Model 2090 programmer. Biventricular pacing (BV), left ventricular free-wall lead (LV), RA pacing and AV sequential pacing to the RV apex lead (RV) were performed in this study. The order of the pacing methods used was random. An AV delay of 30 ms was programmed. After at least 1 minute of pacing, steady state conditions were achieved, during which, 10 seconds of data were collected at each heart rate and pacing site. Then the inferior vena-cava was occluded for 10-12 seconds and data were collected at varying levels of preload (EDV) as well. “During RA pacing, if AV block was encountered, boluses of Glycopyrolate (.01-.04 mg IV) were administered until the block subsided.” [13] After all the data was

collected, the animals were allowed to recover for additional study which is outside the scope of this report.

DATA ANALYSIS:

Tau (τ) was estimated from the data obtained by using the previously discussed four methods. Data were analyzed by a program written using MATLAB™ (Appendix I). The left ventricular pressure LVP was filtered using a fourth-order 50 Hz Butterworth filter. The filter was selected because it has low ripple factor and a high attenuation to filter out the high frequency noise while maintaining the useful pressure signal. The LVP was filtered both causally and then anti-causally to prevent the phase shifts that results from filtering. The filtered data was then differentiated with respect to time (dP/dt) and the peak negative dP/dt was recorded along with the corresponding left ventricular pressure P_0 . A five-point derivative algorithm was used to calculate the derivative. [14] The end diastolic pressure (EDP) of each beat was selected by first finding the indices in the waveform where the dP/dt of the LVP was maximum (P_{max}), then 40% of the P_{max} was found. EDP was found by finding the pressure at the time index 40 ms prior to the index where the pressure was 40% of P_{max} . Nonlinearities at the end beats occur from complexities of performing a VCO in vivo. [15] Due to this, the last 6 beats were excluded for Dog 2 at VCO at pacing site RA with heart rate of 160 as shown in Fig. 3 in Results section. Any incomplete beats at the beginning of the LVP waveform were also excluded for calculation of Tau.

The semi logarithmic model of Tau(τ_L) or τ_1 for the purpose of this study) with zero asymptote of left ventricular pressure is defined as the negative inverse of the slope of the plot of the natural logarithm of left ventricular pressure versus time. This method assumes that the decline of the left ventricular pressure is zero. Therefore pressure waveform, P, is defined as ln

$P = At + B$, where A represents the slope of the line in sec^{-1} and B is the natural log (\ln) of left ventricular pressure at P_0 . [4] By solving this equation at P_0 , τ_1 is estimated to be equal to $-1/A$. The portion of the curve between P_0 and 5mm Hg above the EDP of the previous beat was analyzed by taking the natural log of all the points in the curve and then calculating the slope to find A . $\tau_1(\tau_L)$ was calculated with by finding the negative inverse of A .

The semilogarithmic model using the initial 40ms of isovolumic relaxation, τ_2 or (τ_{40}) was calculated using the same equation except that the portion of the curve analyzed included data points within the first 40ms from P_0 . [5]

The exponential model with non-zero asymptote of left ventricular pressure (τ_E) or τ_3 , does not assume that the fall of the left ventricular pressure is zero. Hence the equation of the pressure is $P(t) = ae^{bt} + c$ where a , b , c are determined iteratively by considering three points $P_{(0)}$, $P_{(m)}$, $P_{(2m)}$ which are equally spaced in time at 20ms intervals between P_0 and EDP of the previous beat using the following equations. [7]

$$b = -1/20 \text{ ms} * \ln[(P_{(2m)} - P_{(m)})/(P_{(m)} - P_{(0)})] \quad (1)$$

$$c = P_{(0)} - a \quad (2)$$

$$a = [(P_{(m)} - P_{(0)}) / (e^{-b*(20\text{ms})} - 1)] \quad (3)$$

$$\tau_E = -1/b \quad (4)$$

In this study $P_{(0)}$ was first assumed to be P_0 and $P_{(m)}$ was equal to $P_0 + 10\text{ms}$ and $P_{(2m)}$ was equal to $P_0 + 20\text{ms}$. Then the next set of pressures was analyzed by adding 1ms to the times 0, m and 2m of the first set of pressures that were taken. These equations were applied successively until all the points in a relaxation event were used. The values for τ_E for each beat were determined as the mean of the τ values of all the segments that were calculated.

The derivative model with non-zero asymptote of left ventricular pressure (τ_D), τ_4 , also does not assume a zero asymptote of left ventricular pressure decline. Raff and Glantz described the ventricular pressure decline by the equation $P = P_0 e^{-t/\tau} + P_B$, where P_B is the additive baseline shift due to pleural or pericardial pressure of measurement errors. [6] Differentiating this equation with respect to time t and then using the equation for P to eliminate $e^{-t/\tau}$ the equation $dP/dt = 1/\tau (P - P_B)$ was obtained. This shows that dP/dt is a linear function of P with a slope of $-1/\tau$. Thus τ_D is calculated by taking the negative inverse of the linear regression of dP/dt against P using all the points between P_0 and EDP of the previous beat.

RESULTS AND DISCUSSIONS

A program was written in Matlab™ in two parts and can be found in Appendix 1. Fig. 1 represents the flow chart of the program. The first part named importer2.m loads the original waveform data from Excel™ sheets in to Matlab™ and the second part titled dpdtmin.m analyses the data to calculate τ .

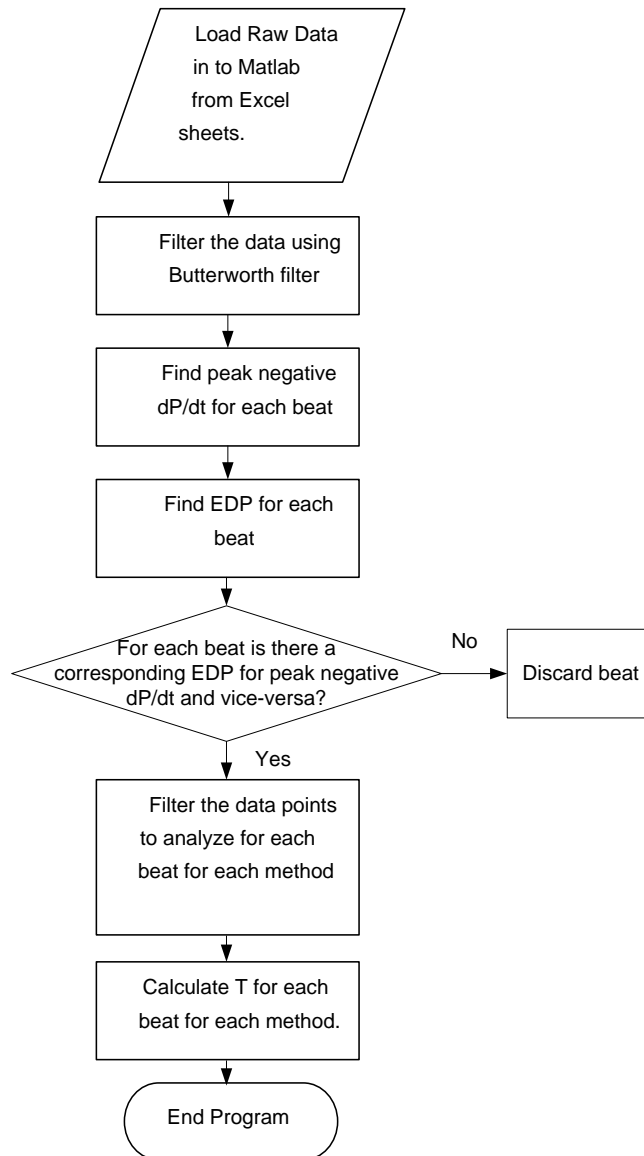


Fig.1. Flow-diagram of the algorithm of the program used.

Next, physiological waveform data from 80 Excel® sheets were loaded into the program successfully. The data was filtered and shifted to positive quadrant to address negative pressures. This did not affect the calculation of the time constant, τ .

A sample waveform of an unfiltered LVP is shown in Fig. 2. It can be seen from this figure that the LVP waveform is noisy. To remove this noise, a 50 Hz Butterworth filter was implemented. Note also, the declining LVP due to the vena caval occlusion.

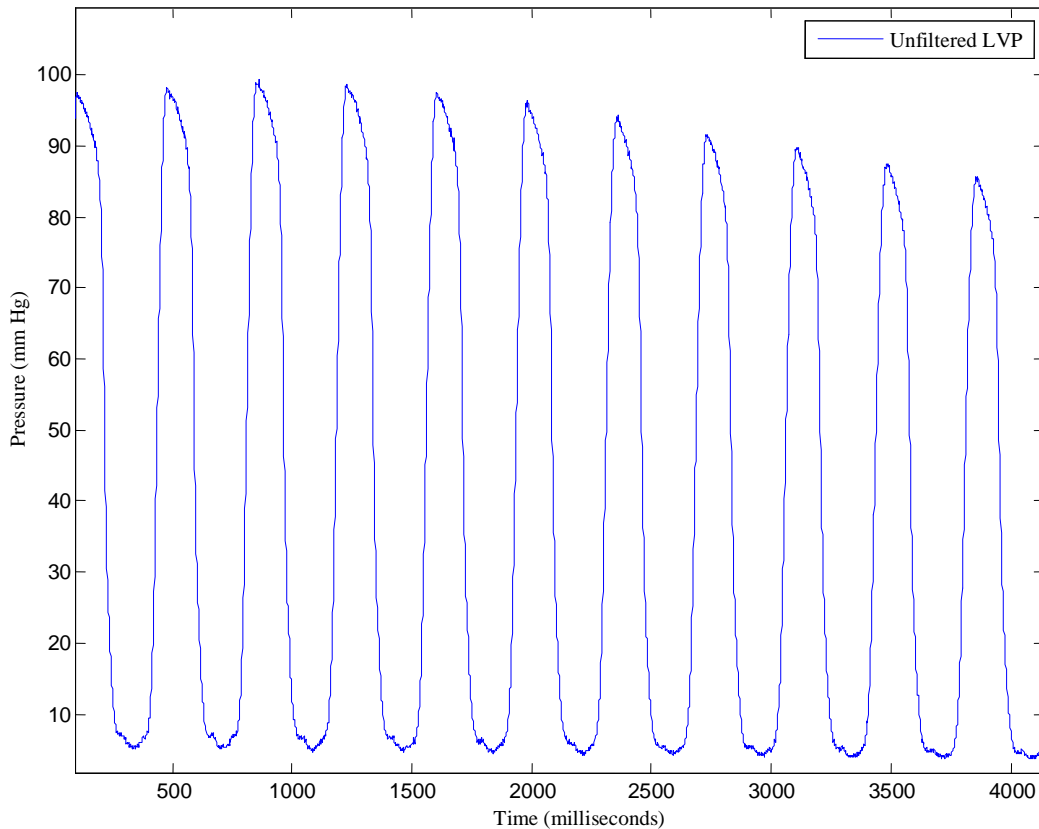


Fig.2. Sample waveform of unfiltered LVP.

Fig.3 shows the waveform after a Butterworth filter was applied to it. This figure shows how the filter has removed the noise to output a better signal. The filtered waveform was also shifted to the positive quadrant since in VCO, negative pressures can be encountered. Negative pressures affect the calculation of τ especially in the exponential model where the difference of pressures is used in calculations.

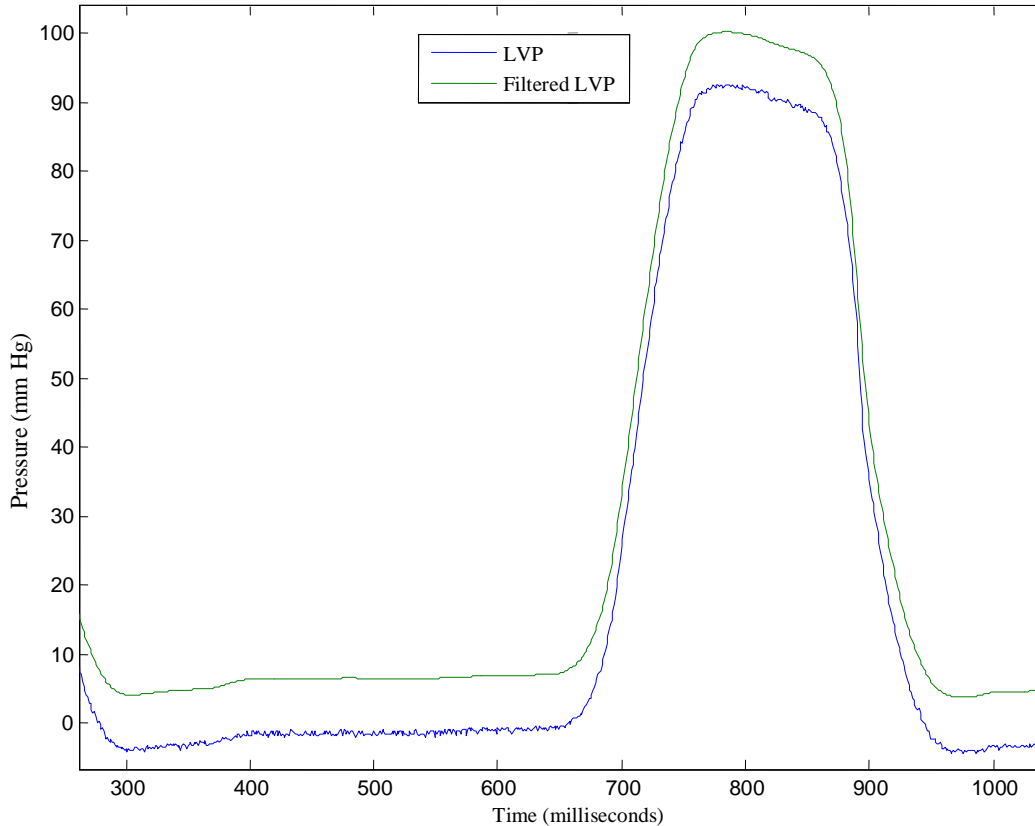


Fig.3. Plot of LVP Vs filtered and offset LVP.

The peak negative dp/dt was calculated for each beat as was the EDP. Some beats were incomplete and an EDP for the previous beat could not be calculated. These beats were left out of the analysis in the program. Similarly, some beats had EDP which did not have a corresponding peak negative dp/dt . These beats were left out of the program as well. In Dog 2, at pacing site R.A. with a heart rate of 160 bpm (see Fig. 4), the last 6 beats were erratic and an EDP was difficult to quantify. Hence these beats were also removed from the analysis. This waveform quality check was important to ensure that the data analyzed was comprised of complete beats in a normal VCO state.

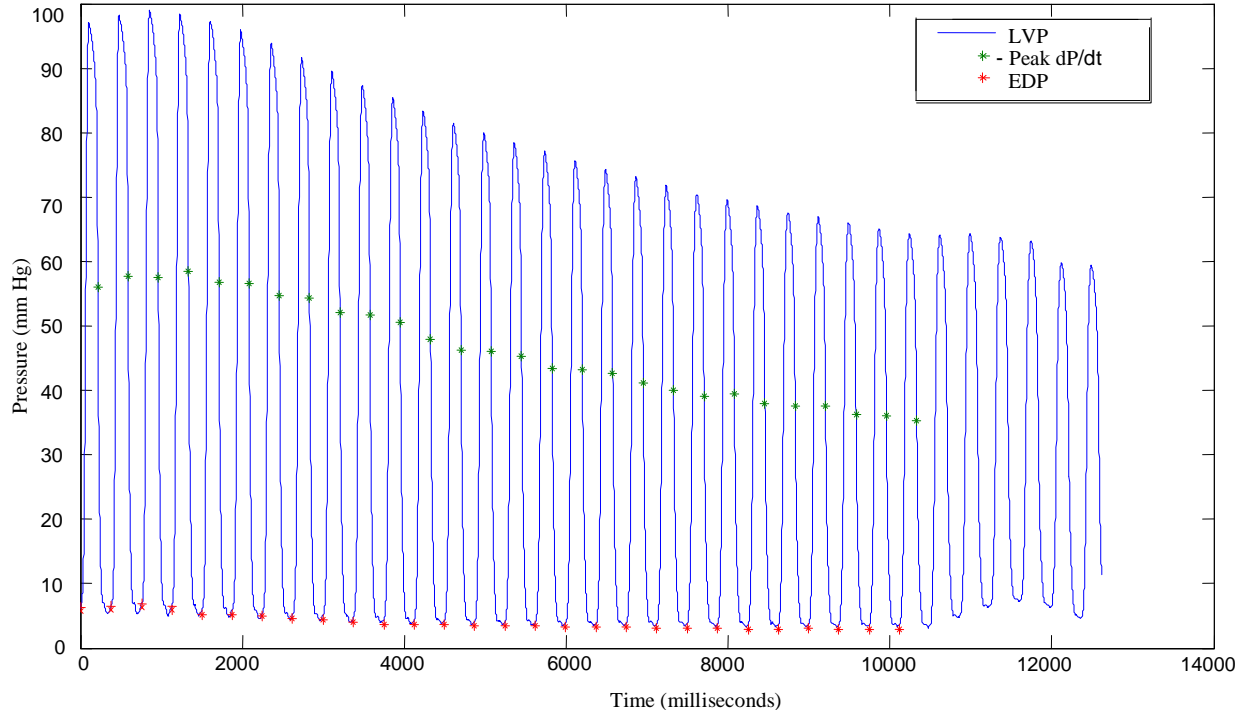


Fig.4. The LVP waveform of Dog 2 at VCO at pacing site R.A. with heart rate of 160 bpm with EDP and negative peak dP/dt marked.

To validate the algorithm used, data was synthesized with τ and other constants set to a known value and then the program was used to calculate τ from the synthesized data. Table 1 shows the results of the validation.

Table.1. Validation of the algorithm used.

Method	Equation used	τ_{set}	τ_{obtained}	Other variables set
1	$\ln P = At + b$.434 s	.434 s	$b = 4.62$ $A = -2.3$
2	$\ln P = At + b$.074 s	.073 s	$b = 4.62$ $A = -13.54$
3	$P(t) = ae^{bt} + c.$.0076 s	.0075 s	$a = 0.2170$ $b = -0.131$ $c = 6.83$
4	$dP/dt = -(P_0/T)*e^{-t/T}$	0.022s	.022 s	$P_0 = 55.98$

The equation constants and τ were set to random values. For each method, synthesized data was obtained using known values of τ and other constants. The synthesized data was then analyzed and the τ calculated was compared with the known τ . As seen in Table 1, for each model, the τ calculated was approximately equal to the τ set to a known value to synthesize the data indicating that the program was accurate in calculating τ . Fig. 5 confirms this with the plot of the curve fit of the synthesized data. In all cases the program's τ was the same as the synthesized τ . The maximum error in estimating τ from synthesized data was not greater than 0.1%

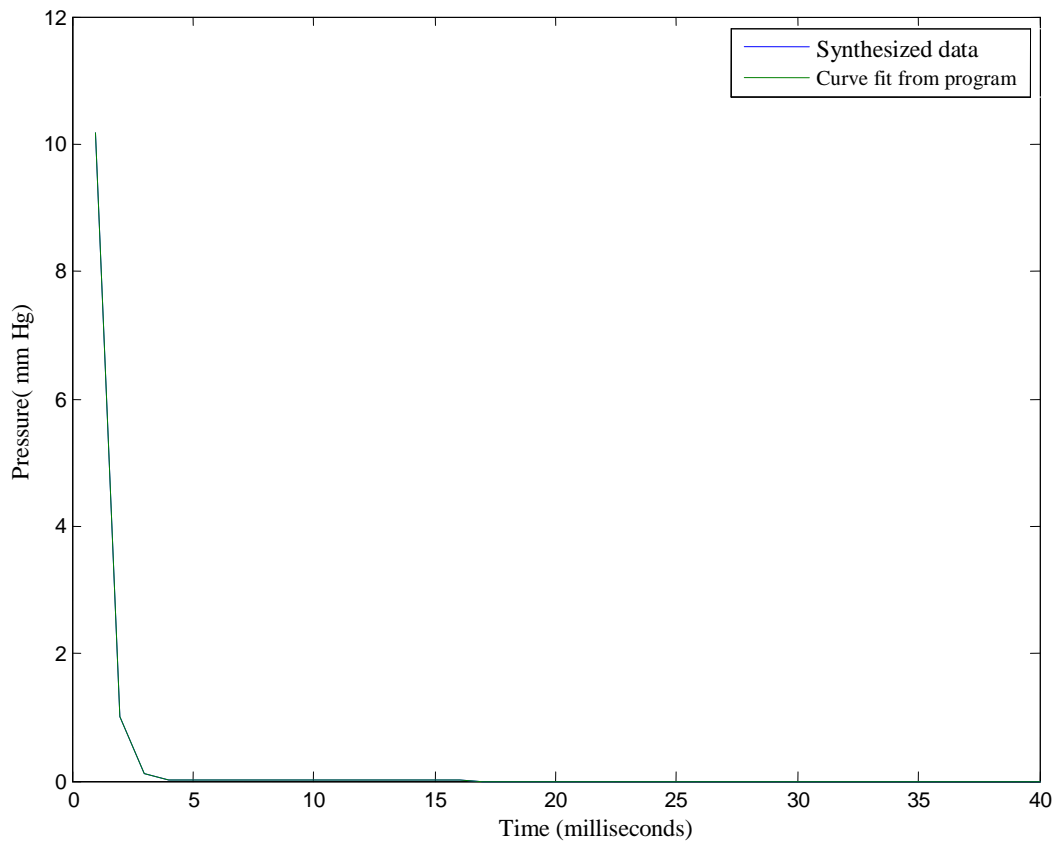


Fig.5. Plot of synthesized data.

Once the program was validated, it was used on physiological data. Fig.6 shows a single heart beat with different portions marked for analysis of τ using different methods. The data points between peak negative dP/dt and the point corresponding to EDP +5mm Hg in isovolumic relaxation were used for calculating τ using semi logarithmic method. The data points for 40 ms after peak negative dP/dt were used to calculate τ using method 2. For the exponential and derivative models of τ , the data points between peak negative dP/dt and the point corresponding to the EDP in isovolumic relaxation were analyzed.

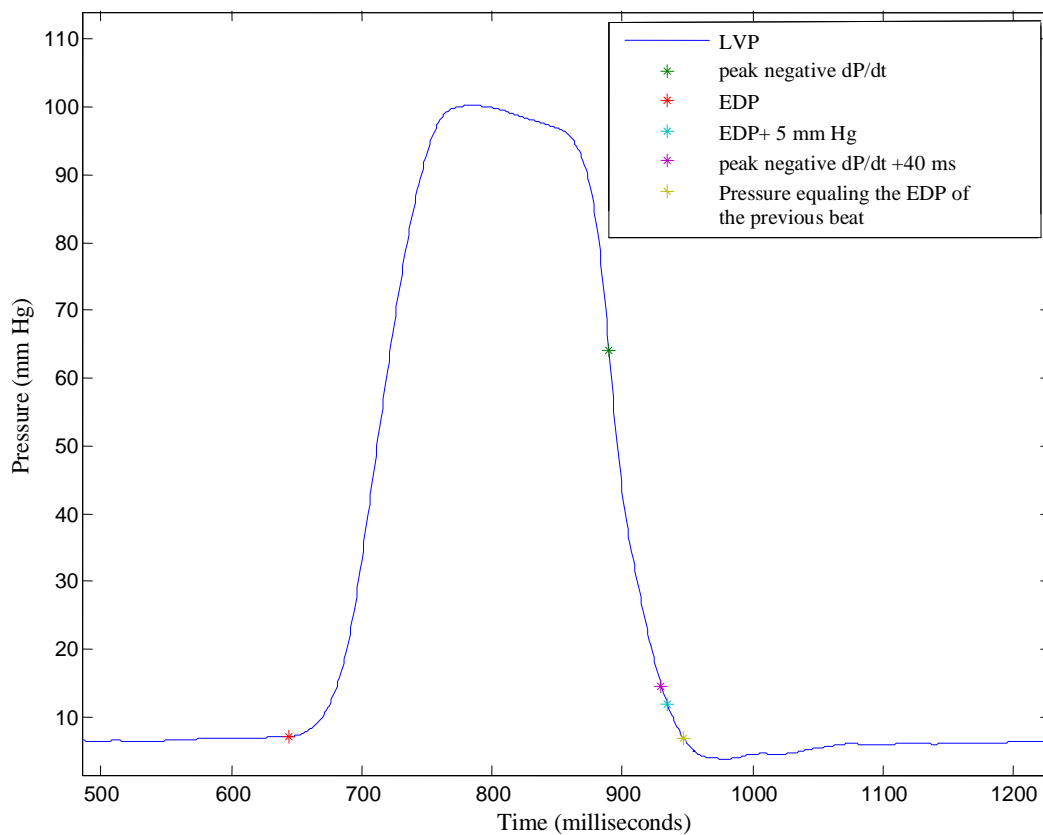


Fig.6. A single heartbeat with portions used for analysis of different methods.

Fig.7 shows the τ that was calculated using the four methods for a representative VCO data set of heart beats.

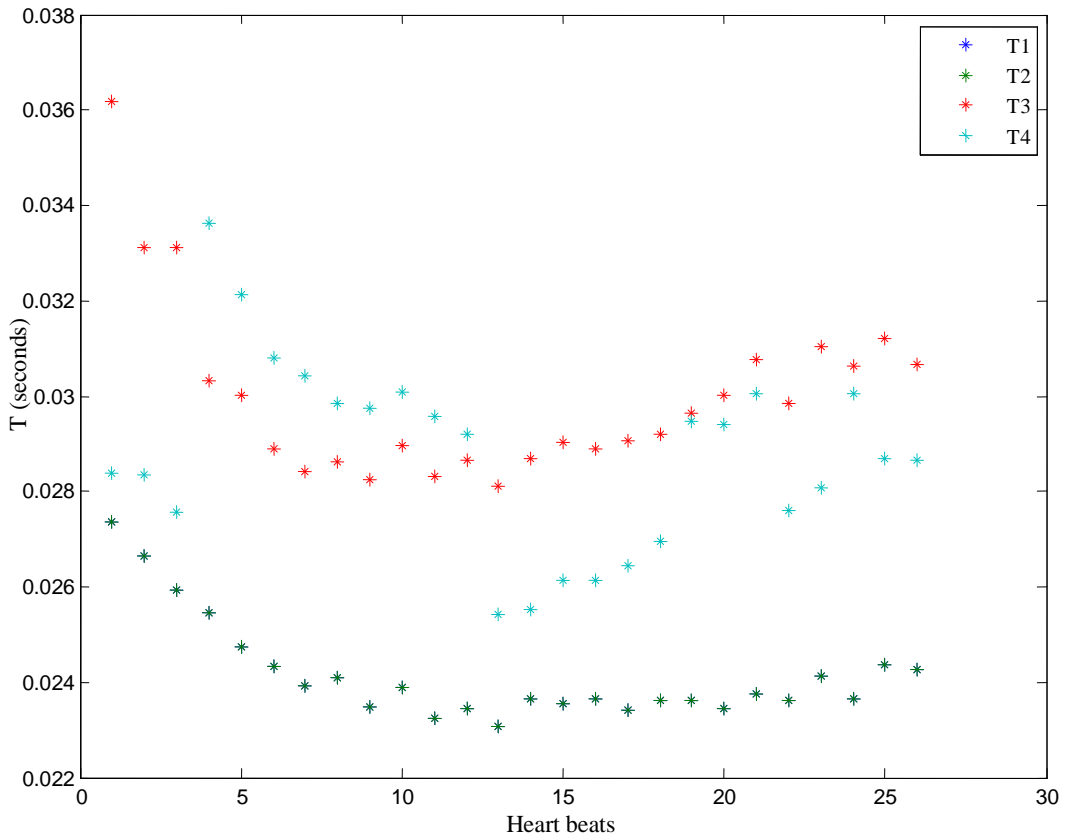


Fig.7. Comparison of τ calculated by four methods for Dog 1 at pacing site BV at a rate of 90 BPM during VCO.

τ calculated using the semi logarithmic models are identical as seen in this figure. It is also relatively smaller in value compared to τ calculated using exponential and differential model. This is probably due to the assumption of zero asymptote of fall of left ventricular pressure inherent in these semi logarithmic models. τ calculated using the exponential model is seen to be the greater than τ calculated using other models across the beats during VCO. This indicates that there is difference in τ calculated using different models.

Fig.8 shows the curve fits for the LVP analyzed, using the four methods.

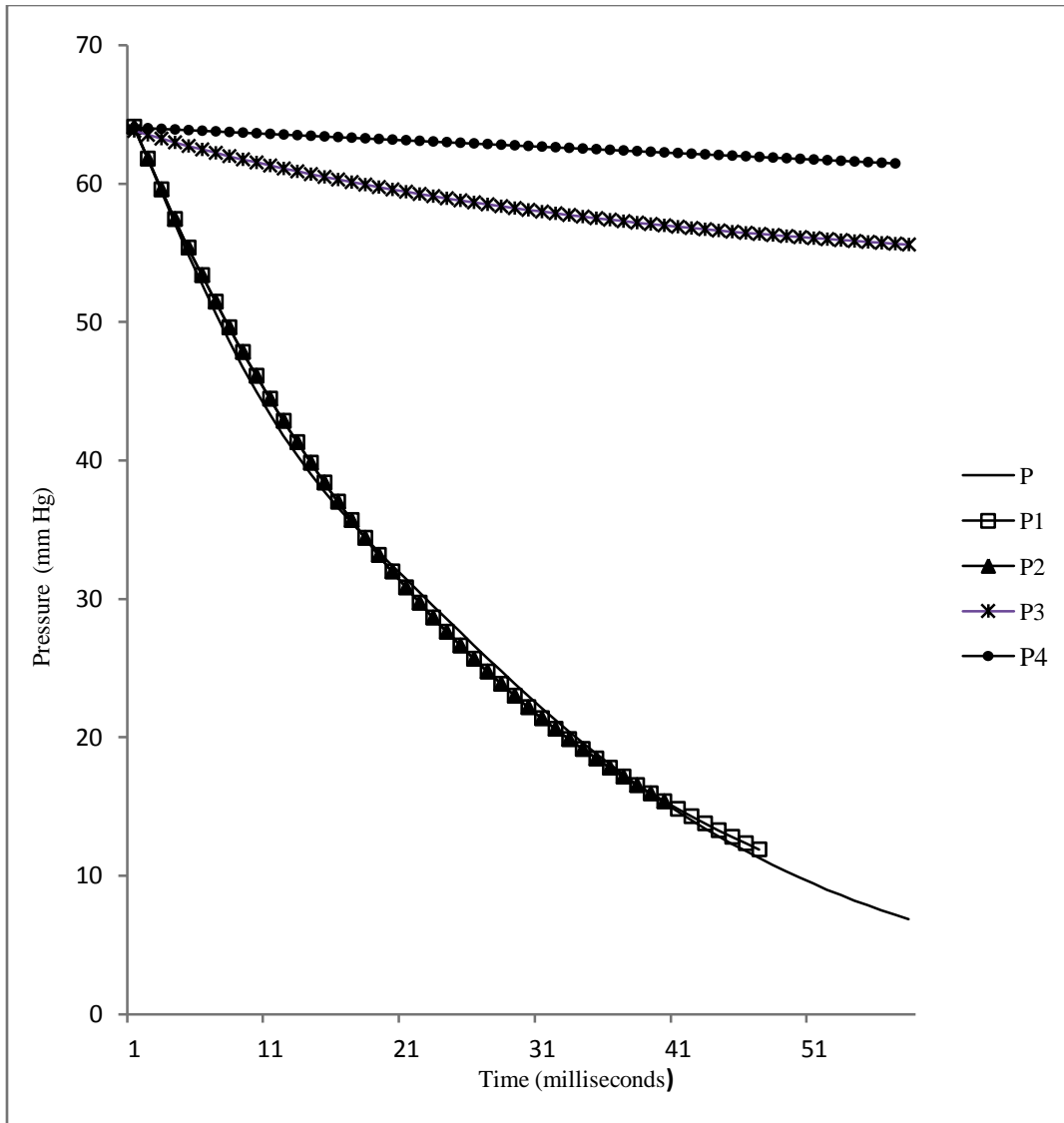


Fig.8. Curve fits for the LVP analyzed, using the four methods.

The curve represented by P is the portion of the isovolumic relaxation that was analyzed. P1 represents the curve fit of P using the model proposed by Weiss et al. [4] P2 represents the curve fit of P using the model proposed by Rousseau et al. [5] P3 represents the curve fit of P using the model proposed by Thompson et al. [7] P4 represents the curve fit of P using the model proposed by Raff and Glantz. [6] This figure shows how the different models vary in fitting the curve of LVP that is analyzed. The curve fits using the semi-logarithmic models are very similar

since these models use the same equation to calculate τ . From this figure, curve fit using the semi logarithmic models seems to mimic the LVP curve the best.

The τ calculated using this program was compared with that found by Raff and Glantz in their paper, "Volume loading slows left ventricular isovolumic relaxation rate. Evidence of load-dependent relaxation in the intact dog heart." *Circulation Research*, 1981: 48:813-824. The range of τ observed in this study was around 24 msec to 44.5 msec. The range of τ as seen in Fig.7 is from 23 msec to 36 msec. The similarity in the results of these two studies indicate that the τ estimated was in the appropriate range for the animal and hence successfully estimated.

This program calculates the τ for 5 dogs at 4 pacing sites, and at two heart rates during VCO. The program is flexible and τ can be calculated using different permutations and combinations. This analysis is useful in determining if τ can be used as a diagnostic to estimate the optimal site to pace the heart. Also the effect of heart rate on τ can be determined. Hence this program is an effective tool in the analysis of T from the raw LVP data derived.

CONCLUSION

A program was developed that calculates LVP relaxation time constant, τ , from physiological data. The code was verified over synthesized data. This program calculates the τ for 5 dogs at 4 pacing sites, and at two heart rates during VCO. Using this program, τ for different dogs at different pacing sites while pacing at different heart rates can be calculated and analyzed. τ can be found for all the beats in a Dog or for each beat. The program is flexible and τ can be calculated using different permutations and combinations. This analysis is useful in determining if τ can be used as a diagnostic to estimate the optimal site to pace the heart. Also the effect of heart rate on τ can be determined. Hence this program is an effective tool in the analysis of τ from the raw LVP data derived.

REFERENCES

- [1] O. H. L. Bing, J. F. Keefe, M. J. Wolk, L. J. Finkelstein and H. J. Levine. "Tension prolongation during recovery from myocardial hypoxia." *The American Society for Clinical Investigation*, vol. 50, pp. 660-666, 1971.
- [2] J. V. Tyberg, L. A. Yeatman, W. W. Parmley, C. W. Urschel, and E. H. Sonnenblick. "Effects of hypoxia on mechanics of cardiac contraction." *American Journal of Physiology*, vol. 218 no. 6, pp.1780-1788, 1970.
- [3] J.W. Frederiksen, J. L. Weiss, and M. L. Weisfeldt. "Time constant of isovolumic pressure fall: determinants in the working left ventricle." *American Journal of Physiology*, vol. 235 no. 6, pp. H701-H706, 1978.
- [4] J. L. Weiss, J. W. Frederiksen, and M. L. Weisfeldt. "Hemodynamic determinants of the time-course of fall in canine left ventricular pressure." *The Journal of Clinical Investigation*, vol. 58 no. 3, pp. 751-760, 1976.
- [5] M. F. Rousseau, C. Veriter, J.M. Detry, L. Brasseur and H. Pouleur. "Impaired early left ventricular relaxation in coronary artery disease: effects of intracornary nifedipine." *Circulation*, vol. 62, pp. 764-772, 1980.
- [6] G. L. Raff and S. A. Glantz. "Volume loading slows left ventricular isovolumic relaxation rate. Evidence of load-dependent relaxation in the intact dog heart." *Circulation Research*, vol. 48, pp. 813-824, 1981.
- [7] D. S. Thompson, C. B. Waldron, S. M. Juul, N. Naqvi, R. H. Swanton, D. J. Coltart, B. S. Jenkins, and M. M. Webb-Peploe. "Analysis of Left Ventricular Pressure During Isovolumic Relaxation in Coronary Artery Disease." *Circulation*, vol. 65, pp. 690-697, 1982.
- [8] P. D. Bourdillon, B. H. Lorell, I. Mirsky, W. J. Paulus, J. Wynne and W. Grossman. "Increased regional myocardial stiffness of the left ventricle during pacing-induced angina in man." *Circulation*, vol. 67, pp. 316-323, 1983.
- [9] P. W. Serruys, W. Wijns, M. Van Den Brand, S. Meij, C. Slager, J. C. H. Schuurbiers, P. G. Hugenholtz, and R. W. Brower. "Left ventricular performance, regional blood flow, wall motion, and lactate metabolism during transluminal angioplasty." *Circulation*, vol. 70, pp. 25-36, 1984.
- [10] M. R. Starling, D. G. Montgomery, G. B. John Mancini, and R. A. Walsh. "Load independence of the rate of isovolumic relaxation in man." *Circulation*, vol. 76, pp. 1274-1281, 1987.

- [11] M. V. T. Tantengco, R. L. Thomas and P. P. Karpawich. "Left ventricular dysfunction after long-term right ventricular apical pacing in the young." *Journal of the American College of Cardiology*, vol. 37, pp. 2093-2100, 2001.
- [12] T. M. Kolettis, Z. S. Kyriakides, D. Tsiapras, T. Popov, I. A. Paraskevaides and D. T. Kremastinos. "Improved Left Ventricular Relaxation During Short-term Right Ventricular Outflow Tract Compared to Apical Pacing." *CHEST*, vol. 117, pp. 60-64, 2000.
- [13] A. Black, N. Grenz, N. Schaible, P. Arndt, J. Lucht, K. Nesvig, D. Ewert and L. Mulligan. "Assessment of $d\sigma^*/dt_{max}$, a Load Independent Index of Contractility, in the Canine." *Cardiovascular Engineering*, vol. 9, pp. 49-55, 2009.
- [14] A. E. Marble, C. M. McIntyre, R. Hastings-James, C. W. Hor. "A comparison of digital algorithms used in computing the derivative of left ventricular pressure." *IEEE Trans Biomed Eng.*, vol. 28 no. 7, pp. 524-529, 1981.
- [15] D. Burkhoff, S. Sugiura, D. T. Yue, and K. Sagawa. "Contractility-dependent curvilinearity of end-systolic pressure-volume relations." *AJP - Heart*, vol. 252 no. 6, pp. H1218-H1227, 1987.

APPENDIX I

Program in MATLAB™:

```
%”importer.m”
```

```
global animal;
```

```
%change to the correct directory
```

```
dogs = {'329938','329951','329948','330644','330634'};
```

```
loads = {'SS','VCO'};
```

```
sites = {'BV','LV','RA','RV'};
```

```
rates = {'90','160'};
```

```
for dog=1:length(dogs)
```

```
    for load=1:length(loads)
```

```
        for site=1:length(sites)
```

```
            for rate=1:length(rates)
```

```
                filename = strcat(dogs(dog),'_',sites(site),rates(rate),'_',loads(load));
```

```
                filename = char(filename);
```

```
                matrix(['m',filename]) = xlsread(filename);
```

```
                animal(dog,load,site,rate).time=matrix(['m',filename])(:,1);
```

```
                animal(dog,load,site,rate).LVP=matrix(['m',filename])(:,2);
```

```
                animal(dog,load,site,rate).Vtot=matrix(['m',filename])(:,3);
```

```
                animal(dog,load,site,rate).V1=matrix(['m',filename])(:,4);
```

```
                animal(dog,load,site,rate).V2=matrix(['m',filename])(:,5);
```

```
                animal(dog,load,site,rate).V3=matrix(['m',filename])(:,6);
```

```
                animal(dog,load,site,rate).V4=matrix(['m',filename])(:,7);
```

```
                animal(dog,load,site,rate).V5=matrix(['m',filename])(:,8);
```

```
                animal(dog,load,site,rate).EGMPV=matrix(['m',filename])(:,9);
```

```

animal(dog,load,site,rate).ECG1=matrix(['m',filename])(:,10);
animal(dog,load,site,rate).EGMRV=matrix(['m',filename])(:,11);
animal(dog,load,site,rate).EGMLV=matrix(['m',filename])(:,12);
animal(dog,load,site,rate).Marker=matrix(['m',filename])(:,13);
animal(dog,load,site,rate).AoP=matrix(['m',filename])(:,14);
if (dog>=4)
    animal(dog,load,site,rate).ECG2=matrix(['m',filename])(:,15);
    animal(dog,load,site,rate).ECG3=matrix(['m',filename])(:,16);
else
    animal(dog,load,site,rate).CF=matrix(['m',filename])(:,15);
    animal(dog,load,site,rate).ECG2=matrix(['m',filename])(:,16);
    animal(dog,load,site,rate).ECG3=matrix(['m',filename])(:,17);
end
end
end
end
end
end
%“dpdt.min”
load globalanimal2009.mat
fs = 1000;
[b,a] = butter(4,40/(1000/2)); % Create a butterworth filter
for j1= 6:10:76
for i = j1:j1+4
    animal(i).LVP_Filt = filtfilt(b,a,animal(i).LVP); % Apply Butterworth filter to the LVP
    waveform

```

```
animal(i).q=min(animal(i).LVP_Filt);% Shifting the waveform in to the positive quadrant.  
since VCO is used and some of the waveforms can run in to negative pressures.
```

```
if animal(i).q<0
```

```
    animal(i).LVP_Filt1=animal(i).LVP_Filt+abs(animal(i).q);
```

```
    else
```

```
        animal(i).LVP_Filt1=animal(i).LVP_Filt;
```

```
    end
```

```
    animal(i).DLVP_Filt = diff(animal(i).LVP_Filt1).*fs;% Differentiating the filtered LVP
```

```
    animal(i).DLVP_Filt1 = diff(animal(i).LVP_Filt).*fs;% Differentiating the normalised  
    filtered LVP
```

```
    animal(i).DLVP_Filt2 = diff(animal(i).DLVP_Filt1).*fs;% Double differentiating the  
    normalised filtered LVP
```

```
    animal(i).DLVP = diff(animal(i).LVP).*fs;% Differentiating the LVP.
```

```
    animal(i).maxvector=0; % Initialising maxvector
```

```
    animal(i).minvector=0; % Initialising minvector.
```

```
    animal(i).LVP_Filtminindices=0;% Initialising LVP_Filtminindices
```

```
    clf
```

```
    animal(i).maxpoints = find(animal(i).DLVP_Filt>=.4*max(animal(i).DLVP_Filt)); % Finding  
    all point greater than midway point in the differentiated LVP waveform to find the dp/dt max  
    point.
```

```
    animal(i).maxjump = find(diff(animal(i).maxpoints)~=1); % Finds points in the positive  
    part of the waveform.
```

```
    animal(i).maxindices = animal(i).maxpoints(animal(i).maxjump); % Finds the indices of  
    the points.
```

```
    animal(i).minpoints = find(animal(i).DLVP_Filt<=0.5*max(animal(i).DLVP_Filt)); %  
    Finding all point lower than midway point in the differentiated LVP waveform to find the  
    dp/dt min point.
```

```
    animal(i).minjump = find(diff(animal(i).minpoints)~=1); % Finds points in the negative  
    part of the waveform
```

```

    animal(i).minindices = animal(i).minpoints(animal(i).minjump);% Finds the indices of
    the points.

% find dpdt max indices

for r=1:length(animal(i).maxindices)+1

    if r <=length(animal(i).maxindices)-1

        [animal(i).dpdtmax,animal(i).dpdtmaxindex]=max(animal(i).DLVP_Filt(animal(i).maxi
        ndices(r):animal(i).maxindices(r+1)));% Finds the maximum value of dp/dt of LVP and
        the index at which it occurs between the two points.

        animal(i).dpdtmaxindex=animal(i).dpdtmaxindex+animal(i).maxindices(r)-1;

    elseif r==length(animal(i).maxindices)+1

        [animal(i).dpdtmax,animal(i).dpdtmaxindex]=max(animal(i).DLVP_Filt(1:animal(i).ma
        xindices(1)));

    else

        [animal(i).dpdtmax,animal(i).dpdtmaxindex]=max(animal(i).DLVP_Filt(animal(i).maxin
        dices(r):(length(animal(i).DLVP_Filt))));

        animal(i).dpdtmaxindex=animal(i).dpdtmaxindex+animal(i).maxindices(r)-1;

    end

    animal(i).maxvector(r,1)= animal(i).dpdtmax; % Contains the max value of dP/dt

    animal(i).maxvector(r,2)= animal(i).dpdtmaxindex;% Contains the index at which max
    dP/dt occurs.

end

animal(i).maxvector = sort(animal(i).maxvector,1,'ascend');

if i==67

    for r=1:length(animal(i).maxvector)-6

        animal(i).maxvector2(r,2)=animal(i).maxvector(r,2); % Removing the last 6 beats since
        they are unstable.

    end

elseif (animal(i).maxvector(1,2)<100) % Removing incomplete beats where edp of previous
cannot be measured.

```

```

for r=1:(length(animal(i).maxvector)-1)
    animal(i).maxvector2(r,1)=animal(i).maxvector(r+1,1);
    animal(i).maxvector2(r,2)=animal(i).maxvector(r+1,2);
end
else
    for r=1:(length(animal(i).maxvector))
        animal(i).maxvector2(r,2)=animal(i).maxvector(r,2);
    end
end
end
% Find dpdt min indices
for u=1:length(animal(i).minindices)+1
    if u<=length(animal(i).minindices)-1
        [animal(i).dpdtmin,animal(i).dpdtminindex]=min(animal(i).DLVP_Filt(anim
al(i).minindices(u):animal(i).minindices(u+1)));
        animal(i).dpdtminindex=animal(i).dpdtminindex+animal(i).minindices(u)-1;
    elseif u==length(animal(i).minindices)+1
        [animal(i).dpdtmin,animal(i).dpdtminindex]=min(animal(i).DLVP_Filt(1:anim
al(i).minindices(1)));
    else
        [animal(i).dpdtmin,animal(i).dpdtminindex]=min(animal(i).DLVP_Filt(anim
al(i).minindices(u):(length(animal(i).DLVP_Filt))));
        animal(i).dpdtminindex=animal(i).dpdtminindex+animal(i).minindices(u)-1;
    end
end
animal(i).minvector(u,1)=animal(i).dpdtmin; % Contains the min value of dP/dt
animal(i).minvector(u,2)=animal(i).dpdtminindex;% Contains the index at which it
occurs.
end

```



```

animal(i).minvector = sort(animal(i).minvector,1,'ascend');
if i==67
    for r=1:length(animal(i).maxvector)-6
        animal(i).minvector1(r,2)=animal(i).minvector(r,2); % Removing the last 6 beats due
        to instability.
    end
elseif animal(i).maxvector2(1,2)>animal(i).minvector(1,2)
% Removing incomplete beats where edp of previous cannot be measured.
for r=1:length(animal(i).minvector)-1
    animal(i).minvector1(r,2)=animal(i).minvector(r+1,2);
end
else
    for r=1:length(animal(i).minvector)
        animal(i).minvector1(r,2)=animal(i).minvector(r,2);
    end
end
if length(animal(i).maxvector2)>length(animal(i).minvector1)% To ensure complete beats.
    for r= 1:length(animal(i).minvector1)
        animal(i).maxvector1(r,2)=animal(i).maxvector2(r,2);
    end
else
    for r= 1:length(animal(i).minvector1)
        animal(i).maxvector1(r,2)=animal(i).maxvector2(r,2);
    end
end
end

```

`%Find edp indices`

```
for r=1:length(animal(i).maxvector1)
```

```
    animal(i).Pdmax(r,1)= animal(i).LVP_Filt1(animal(i).maxvector1(r,2));
```

```
    animal(i).edpm(r,1) =(0.4*(animal(i).Pdmax(r,1)));
```

```
    e=animal(i).edpm(r,1);
```

```
    f=animal(i).maxvector1(r,2);
```

```
    difference= (animal(i).LVP_Filt1(f)-e);
```

```
    while (difference>0 )
```

```
        f=f-1;
```

```
        difference = (animal(i).LVP_Filt1(f)-e);
```

```
    end
```

```
    animal(i).edpindices(r,1) = f-40;
```

```
    animal(i).edpf(r,1)=animal(i).LVP_Filt1(animal(i).edpindices(r,1)); % End Diastolic  
    pressure
```

```
    animal(i).edpf1(r,1)=((animal(i).LVP_Filt1(animal(i).edpindices(r,1)))+5);
```

```
end
```

```
animal(i).edpindices(r,1)=sort(animal(i).edpindices(r,1),1,'ascend'); %
```

`Indices of the EDP of all the beats.`

```
animal(i).edpf(r,1)=sort(animal(i).edpf(r,1),1,'ascend'); % EDP of beats
```

```
animal(i).edpf1(r,1)=sort(animal(i).edpf1(r,1),1,'ascend');% EDP +5 mm Hg
```

```
animal(i).Pdmax(r,1)=sort(animal(i).Pdmax(r,1),1,'ascend');
```

`% Semilogarithmic model with zero asymptode of left ventricular pressure`

```
for k=1:length(animal(i).minvector1(:,2))
```

```
l=((animal(i).LVP_Filt1(animal(i).minvector1(k,2)))-animal(i).edpf1(k,1)); % finding  
all the data points between P0 and 5 mm Hg above EDP of the previous beat.
```

```
f=animal(i).minvector1(k,2);
```

```
while (l>0) && (f<length(animal(i).LVP_Filt1))
```

```
f=f+1;
```

```
l = (animal(i).LVP_Filt1(f)-animal(i).edpf1(k,1));
```

```
end
```

```
animal(i).edpindices(k,2) = f;
```

```
end
```

```
p=find(animal(i).edpindices(:,2));
```

```
p1 =length(p);
```

```
animal(i).edpindices1(:,2)=(animal(i).edpindices(1:p1,2));
```

```
for k=1:length(animal(i).minvector1(:,2))
```

```
l=((animal(i).LVP_Filt1(animal(i).minvector1(k,2)))-animal(i).edpf(k,1));
```

```
f=animal(i).minvector1(k,2);
```

```
while (l>0) && (f<length(animal(i).LVP_Filt1))
```

```
f=f+1;
```

```
l = (animal(i).LVP_Filt1(f)-animal(i).edpf(k,1));
```

```
end
```

```
animal(i).edpindices(k,3) = f;
```

```
end
```

```
p=find(animal(i).edpindices(:,3));
```

```
p1 =length(p);
```

```
animal(i).edpindices1(:,3)=(animal(i).edpindices(1:p1,3));
```

```
%
```

```

for u=1:length(animal(i).minvector1(:,2))
    j=1;
    animal(i).ppoints(j,u)=(animal(i).minvector1(u,2));
    animal(i).logpoints(j,u)= log(animal(i).LVP_Filt1(animal(i).ppoints(j,u)));
    l=(animal(i).minvector1(u,2));
    for q=((animal(i).minvector1(u,2)):(animal(i).edpindices1(u,2)))
        j=j+1;
        animal(i).ppoints(j,u)= (animal(i).ppoints(j-1,u))+1;
        animal(i).logpoints(j,u)= log(animal(i). LVP_Filt1(animal(i).ppoints(j,u)));
    end
    animal(i).k1=find(animal(i).ppoints(:,u));
    k=length(animal(i).k1);
    animal(i).fitvalues(u,1:2)=polyfit((animal(i).ppoints(1:k,u)*(1/fs)),(animal(i).logpoints(1:k,u)
),1);
    animal(i).tauvalues1(u,1)= (-1)*(1/(animal(i).fitvalues(u,1)));
end

```

%semilogarithmic model using initial 40ms of isovolumic relaxation(T40)

```

for u=1:length(animal(i).minvector1(:,2))
    j=1;
    animal(i).ppoints1(j,u)=(animal(i).minvector1(u,2));
    animal(i).logpoints(j,u)= log(animal(i).LVP_Filt1(animal(i).ppoints(j,u)));
    l=(animal(i).minvector1(u,2));
    for q=((animal(i).minvector1(u,2)):(animal(i).minvector1(u,2)+40))
        j=j+1;

```

```

animal(i).ppoints(j,u)= (animal(i).ppoints(j-1,u))+1;
animal(i).logpoints(j,u)= log(animal(i). LVP_Filt1(animal(i).ppoints(j,u)));

```

```
end
```

```
k1=find(animal(i).ppoints(:,u));
```

```
k=length(k1);
```

```
animal(i).fitvalues2(u,1:2)=polyfit((animal(i).ppoints(1:k,u)*(1/fs))',(animal(i).logpoints(1:k,u))',1);
```

```
animal(i).tauvalues2(u,1)= (-1)*(1/(animal(i).fitvalues2(u,1)));
```

```
end
```

```
% EXPONENTIAL MODEL WITH NON-ZERO ASYMPTOTE OF LEFT VENTRICULAR PRESSURE
```

```
s=1;
```

```
for u=1:length(animal(i).minvector1(:,2))
```

```
    animal(i).epoints1=animal(i).minvector1(u,2);
```

```
    animal(i).epoints2=animal(i).minvector1(u,2)+10;
```

```
    animal(i).epoints3=animal(i).minvector1(u,2)+20;
```

```
    animal(i).P1=animal(i).LVP_Filt1(animal(i).epoints1);
```

```
    animal(i).P2=animal(i).LVP_Filt1(animal(i).epoints2);
```

```
    animal(i).P3=animal(i).LVP_Filt1(animal(i).epoints3);
```

```
for k=1:(animal(i).edpindices(u,3)-animal(i).epoints3)
```

```
    animal(i).bvalue =((-1/10)*log((animal(i).P1-animal(i).P2)/(animal(i).P2-animal(i).P3)));
```

```
    animal(i).tauvaluem(k,1)=(-1/animal(i).bvalue);
```

```
    animal(i).epoints1=(animal(i).epoints1)+1;
```

```
    animal(i).epoints2=(animal(i).epoints2)+1;
```

```
    animal(i).epoints3=(animal(i).epoints3)+1;
```

```
    animal(i).P1=animal(i).LVP_Filt1(animal(i).epoints1);
```

```

    animal(i).P2=animal(i).LVP_Filt1(animal(i).epoints2);
    animal(i).P3=animal(i).LVP_Filt1(animal(i).epoints3);
end

    animal(i).tauvalues3(u,1)=mean(animal(i).tauvaluem(:,1));
end

%Derivative model with non-zero asymptote of left ventricular pressure (Td)
for u=1:length(animal(i).minvector1(:,2))
    j=1;
    animal(i).ppoints4(j,u)=(animal(i).minvector1(u,2));
    animal(i).orgp(j,u)= (animal(i).LVP_Filt1(animal(i).ppoints4(j,u)));
    for q=((animal(i).minvector1(u,2)):(animal(i).edpindices(u,3)-1))
        j=j+1;
        animal(i).ppoints4(j,u)= (animal(i).ppoints4(j-1,u))+1;
        animal(i).orgp(j,u)= (animal(i).LVP_Filt1(animal(i).ppoints4(j,u)));
    end
    animal(i).dpdtp=diff(animal(i).orgp(:,u)).*fs;
    k=length(animal(i).orgp(:,u));
    animal(i).fitvalues(u,1:2)=polyfit((animal(i).orgp(1:(k-1),u)),(animal(i).dpdtp),1);
    animal(i).tauvalues4(u,1)= (-1)*(1/(animal(i).fitvalues(u,1)));
    end
end
end

```