Aus dem Institut für Pharmakologie und Toxikologie
(Prof. Dr. med. W.-H. Zimmermann)
der Medizinischen Fakultät der Universität Göttingen

Development of a Novel Fluorescence Indicator System for the Characterization of Sarcomere Organization and Function in Human Cardiomyocytes

INAUGURAL-DISSERTATION

zur Erlangung des Doktorgrades
der Medizinischen Fakultät der
Georg-August-Universität zu Göttingen

vorgelegt von

Til Driehorst

aus

Göttingen

Göttingen 2021

Dekan:	Prof. Dr. med. W. Brück
Referent:	Prof. Dr. med. W.-H. Zimmermann
Ko-Referent:	Prof. Dr. med. S. E. Lehnart
Drittreferentin:	Prof. Dr. hum. biol. M. Schön

Datum der mündlichen Prüfung: 15.03.2022

Hiermit erkläre ich, die Dissertation mit dem Titel "Development of a Novel Fluorescence Indicator System for the Characterization of Sarcomere Organization and Function in Human Cardiomyocytes" eigenständig angefertigt und keine anderen als die von mir angegebenen Quellen und Hilfsmittel verwendet zu haben.

Göttingen, den …… ……

(Unterschrift)

Table of Contents

List of Figures

List of Tables

Glossary of Acronyms

4Mut	Donor DNA strand termed 4Mut
ACTN2	Alpha-actinin 2
AS	Amino acids
bp	Base pairs
cDNA	Complementary DNA
Citr	Citrine
CM	Cardiomyocytes
CRISPR	Clustered regularly interspaced short palindromic repeats
crRNA	CRISPR RNA
DMEM	Dulbecco's modified Eagle's medium
DNA	Deoxyribonucleic acid
DSB	Double-strand breaks
ΔSL	Change in sarcomere length
ΔSL_{max}	Contraction amplitude
ECM	Extracellular matrix
EDTA	Ethylenediaminetetraacetic acid
EHM	Engineered heart muscle
f	Beating frequency
FACS	Fluorescence activated cell sorting
GFP	Green fluorescent protein
gRNA	Guide RNA
HDR	Homology-directed repair
hESC	Human embryonic stem cells
HES-2	Human embryonic stem cells, line 2
HES-2-ACTN2-Citr	Human embryonic stem cells, line 2, genome-edited for an ACTN2-Citrine fusion protein
HES-2-ACTN2-Citr-CM	HES-2-ACTN2-Citr-derived cardiomyocytes
HES-2-Citr-4Mut	HES-2-ACTN2-Citr stem cell line generated with donor 4Mut
HES-2-Citr-Linker	HES-2-ACTN2-Citr stem cell line generated with donor Linker
HES-2-Citr-SerGG	HES-2-ACTN2-Citr stem cell line generated with donor SerGG
HES-2-WT	Wild-type HES-2
hiPSC	Human induced pluripotent stem cells
hiPSC-CM	Human induced pluripotent stem cell-derived cardiomyocytes
hPSC	Human pluripotent stem cells
hPSC-CM	Human pluripotent stem cell-derived cardiomyocytes
Linker	Donor DNA strand termed Linker

Max. ΔSL_{max}	Maximal contraction amplitude observed during recording time
Min. SL	Minimal sarcomere length observed during recording time
NHEJ	Non-homologous end joining
OM	Omecamtiv mecarbil
PA	Polyacrylamide
PAM	Protospacer adjacent motifs
PBS	Phosphate-buffered saline
PCR	Polymerase chain reaction
PDMS	Polydimethylsiloxane
PEB	Post-exposure bake
Pi	Inorganic phosphate
PLL-g-PEG	Poly(L-lysine)-g-poly(ethylene glycol)
Rel. $STD_i(T)$	Cell-specific standard deviation of T relative to T
Rel. $STD_i(Tmax)$	Cell-specific standard deviation of T_{max} relative to T
rmANOVA	Repeated measures analysis of variance
RNA	Ribonucleic acid
ROI	Region of interest
RPMI	Roswell Park Memorial Institute medium
SerGG	Donor DNA strand termed SerGG
SEM	Standard error of the mean
SL_0	Sarcomere length at rest
Sp	Streptococcus pyogenes
STD	Standard deviation
$STD_i(T)$	Cell-specific standard deviation of T
T	Beating period
t_{max}	Time point of maximum contraction
T_{max}	Time-to-peak contraction
tracrRNA	Trans-activating CRISPR RNA
t_{start}	Starting time point of contraction
YFP	Yellow fluorescent protein

1 Introduction

1.1 Human Stem Cell-derived Cardiomyocytes

Human cardiomyocytes (CM) cannot be reliably extracted from patient biopsies for long-time culture. Through the advent of human pluripotent stem cells (hPSC) – in particular, human embryonic stem cells (hESC; Thomson et al. 1998) and human induced pluripotent stem cells (hiPSC; Takahashi et al. 2007; Yu et al. 2007) – the basis to provide an, in principle, unlimited number of human cardiomyocytes has been established, as hPSC can be differentiated into various tissue cell types, including CM (Itskovitz-Eldor et al. 2000; Kehat et al. 2001; Zhang et al. 2009; Zwi et al. 2009; Burridge et al. 2012). hPSC-derived CM (hPSC-CM) have since created great excitement due to their prospective use in high-throughput pharmacological screening (Braam et al. 2010; Khan et al. 2013; Navarrete et al. 2013; del Álamo et al. 2016; Kurokawa and George 2016), disease modeling (Moretti et al. 2010; Kim et al. 2013; Tiburcy et al. 2017; van Mil et al. 2018), and cardiac replacement therapy (Menasché et al. 2015; Riegler et al. 2015; Tachibana et al. 2017; Oikonomopoulos et al. 2018). However, a major hurdle in their applicability has been, that their level of maturation resembles that of embryonic or fetal rather than adult CM (Robertson et al. 2013; Uosaki et al. 2015; da Rocha et al. 2017). Accordingly, a plethora of research studies have aimed to explore biochemical and biophysical stimuli of cardiac maturation. On the one hand, conventional 2D culture formats have here been demonstrated to lack various of these putative stimuli of maturation, which, in turn, has expedited the development of 3D culture formats such as engineered heart muscle (EHM; Schaaf et al. 2011; Tulloch et al. 2011; Zhang et al. 2013; Tiburcy et al. 2017). On the other hand, ongoing research in advanced 2D culture formats, particularly that of single-cell/oligo-cell CM on defined geometries, aims to decipher further causal nexus between potential maturation stimuli and cardiac function (McDevitt et al. 2002; Bray et al. 2008; Parker et al. 2008; Ribeiro AJS et al. 2015; Ribeiro MC et al. 2015; Boothe et al. 2016).

The degree of hPSC-CM maturation varies strongly between different differentiation protocols, time in culture, culture formats, and even simply among batches of differentiation, but notably, three-dimensional cultures (especially EHM) facilitate a more adult-like CM phenotype, as compared to 2D cultures (Tiburcy et al. 2017; Ronaldson-Bouchard et al. 2018). In the context of this thesis, it is noteworthy to mention several hPSC-CM cell features that reflect the premature phenotype: hPSC-CM are significantly smaller than adult human CM and sarcomere lengths are considerably shorter. Cell spread area and cell volume in hPSC-CM have been reported to be in the range of 500 – 1700 μm^2 and 5000 – 12000 μm^3, respectively, whereas cell volumes of 20000 – 35000 μm^3 have been reported in adult human CM (Severs 2000; Yang et al. 2014; Tiburcy et al. 2017). Sarcomere rest length,

i.e. the rest length of the basic functional unit of cardiac contractile elements (Section 1.3), is typically between 1.6 and 1.9 μm in hPSC-CM, while it is roughly 2.2 μm in adult human CM (Severs 2000; Lundy et al. 2013; Yang et al. 2014; Tiburcy et al. 2017). Furthermore, calcium handling and excitation-contraction coupling are fairly well-developed in hPSC-CM, however, remain slower than in adult CM. hPSC-CM possess few to no T-tubules, while adult CM show an abundance of T-tubles aligned with Z-lines (Section 1.3), and hPSC-CM have smaller calcium stores as compared to adult CM (Yang et al. 2014; Tiburcy et al. 2017; Ronaldson-Bouchard et al. 2018). Moreover, a chronotropic and lusitropic response to β-adrenergic stimulation has been reported, however, an appropriate inotropic response lacks in most hPSC-CM cultures (Robertson et al. 2013; Yang et al. 2014; Tiburcy et al. 2017; Ronaldson-Bouchard et al. 2018).

1.2 The Extracellular Matrix and Its Effects on Cardiomyocyte Function

It is well-recognized, that the cellular microenvironment largely contributes to the maturation and function of cardiomyocytes. In the heart, this microenvironment comprises of cardiomyocytes, non-myocytes, the extracellular matrix (ECM), and paracrine factors. Together, these cellular and extracellular factor constitute interdependent biochemical, biophysical, and bioelectrical cues of what is termed the cardiac or cardiogenic niche, and it is this niche that essentially controls cardiac development, function, and disease (Christalla et al. 2012; Atmanli and Domian 2017).

In the context of this thesis, the ECM and how its mechanical properties dictate CM contractility are of particular interest. Although being mainly recognized as a physical scaffold supporting contraction and relaxation of CM, the cardiac ECM is a highly dynamic entity that facilitates and modulates the above-mentioned biochemical, biophysical, and bioelectrical signaling cues (Atmanli and Domian 2017). It comprises *i)* structural proteins, *ii)* adhesive proteins, and *iii)* proteoglycans (Christalla et al. 2012). The structural proteins are primarily collagen I and III, adhesive proteins include laminin and fibronectin, and altogether, this network of fibers is embedded in a hydrogel-like environment of proteoglycans, such that the ECM components provide a relaxed, stress-resistant meshwork (Frantz et al. 2010). Myocardial cells can communicate directly with the ECM via integrins, that specifically bind to ECM-binding sites. In CM, integrins link to the mechanotransducing costamere complexes, that in turn laterally link to the CM's Z-discs (Section 1.3). Notably, several ECM components, particularly laminin and proteoglycans, are capable of binding various growth factors, indicating the role of the ECM in the regulation of biochemical signaling cues (Christalla et al. 2012; Atmanli and Domian 2017).

While the ECM, as part of the cardiac niche, has been identified as a fundamental building block of CM maturation and function (Christalla et al. 2012; Atmanli and Domian 2017), our understanding of cell-ECM interactions as well as how cell-cell interactions are influ-

enced by the ECM is still limited. Further knowledge of these interactions is, however, crucial, as a deliberate *in-vitro* modeling of the ECM is key to support organotypic differentiation and function of PSC-CM. Accordingly, this *in-vitro* modeling of the ECM has been a major focus in both, 2D and 3D cardiac engineering. 3D strategies have shown great progress in the tuning of an ansiotropic ECM and niche, and have thus generated cardiac constructs with CM that display adult-like properties (Tiburcy et al. 2017; Ronaldson-Bouchard et al. 2018). 2D constructs, however, remain strongly limited due to various constraints. These include *i)* flatness of cells/CM with resulting apical-base polarity, whereas physiologically, CM are rod-like (Baker and Chen 2012); *ii)* unphysiological cell shape and morphology, as compared to the roughly 7:1-shaped adult CM *in vivo* (Gerdes 2002); and *iii)* missing cell-cell and cell-ECM interactions, whereas *in vivo*, CM are embedded in the cardiac niche including cell-cell contacts to roughly eleven other CM (Atmanli and Domian 2017). Furthermore, the layer of adsorbed protein in typical 2D culture formats is in the nanometer range, though cells can sense the environmental stiffness up to 5 μm deep, suggesting that in conventional 2D culture, CM contract against the very stiff culture substrate (Atmanli and Domian 2017).

Despite these limitations, advanced 2D culture formats are still valuable to the cardiovascular research community, not only due to relatively simple experimental protocols, but also because the above-described shortcomings can be selectively remediated, and these tuned 2D cultures then provide platforms to decipher defined structure-function relationships. With respect to matrix elasticity, several significant studies have been conducted in such advanced 2D culture formats, and have mapped correlations between matrix rigidity and CM contractility. Employing traction force microscopy, Engler et al. (2008) examined the contractility of embryonic quail and chicken CM on polyacrylamide (PA) gels of various stiffnesses, and found that substrates that mimic the physiological elasticity of the heart (10-15 kPa; Berry et al. 2006) promoted optimal transmission of contractile work and actomyosin striation, and further facilitated continuous, rhythmic beating. On stiffer substrates with Young's moduli close to that of post-infarct fibrotic scar tissue (3-4 times higher than in the healthy heart; Berry et al. 2006), however, the authors observed a dramatic loss in contractile function including an entire arrest of beating over time. In a similar study employing traction force microscopy, Hazeltine et al. (2012) found that contraction stress of hPSC-CM increased with stiffness of the underlying PA gels in a range from 4-100 kPa, and this effect was not linked to morphology or beating rate. Ribeiro AJS et al. (2015) further showed that hPSC-CM on PA gels of physiological stiffness did not only display improved contractile activity, but also improved calcium flow, mitochondrial organization, electrophysiological properties, and T-tubule formation, when compared to hPSC-CM on stiffer or softer substrates. Notably, in this study, cells were micropatterned to 3:1 - 7:1 rectangular shapes, and the translation of sarcomeric shortening to mechanical output was observed to be highest on 7:1-shaped hPSC-CM.

In several other studies employing techniques of microcontact printing, cell shape had previously been recognized as one of the key regulators of myofibrillar organization and CM function (McDevitt et al. 2002; Bray et al. 2008; Parker et al. 2008; Kuo et al. 2012). McDevitt et al. (2002) micropatterned neonatal rat CM onto 5-50 μm wide laminin lanes, and observed strikingly improved myofibrillar organization, as compared to unpatterned CM. Parker et al. (2008) demonstrated, that neonatal rat CM assumed the geometry of variously shaped fibronectin islands, and formed predictable myofibrillar networks, accordingly. Similarly, Bray et al. (2008) micropatterned neonatal rat CM onto rectangular fibronectin islands with aspect ratios ranging from 1:1 to 7:1, and found that sarcomere alignment was highest at 7:1 aspect ratio. The same group further analyzed the shape dependency of CM contractility, and showed optimized contractility of neonatal rat CM at 7:1 aspect ratios, as measured by systolic stress (Kuo et al. 2012). Parker and colleagues were, moreover, able to link CM shape, ECM elasticity, and CM contractility, when they demonstrated that 7:1-shaped neonatal rat CM generated the most work on physiological 13 kPa PA gels, however, on stiffer 90 kPa substrates, lower aspect ratios (2:1) supported optimal work production (McCain et al. 2014).

1.3 Alpha-Actinin 2 and Its Role as Part of the Contractile Apparatus

The contractile apparatus in cardiomyocytes essentially comprises of a dynamic protein network which is assembled into contractile sarcomeres. These sarcomeres form the basic functional units of cardiac myofibrils. Out of the numerous proteins that constitute and interact with a single sarcomere, perhaps actin and myosin are the most important, as they are the key players in the cardiac contractile cycle. Actin and associated proteins form the polar *thin* filaments, whereas the cardiac muscle isoforms of myosin II form the bipolar *thick* filaments. Actin, at its plus end, is attached to a third key component of the sarcomere, the so-called Z-disc, which is located at both ends of each sarcomere (Figure 1-1). The minus ends of actin thin filaments extend towards the center of the sarcomere, where they overlap with the thick filaments. Upon myosin power strokes (Section 1.4), the thick filaments slide past the thin filaments, which causes the entire sarcomere and thus myofilament and myocyte to contract.

The Z-disc[1] thus cross-links two adjacent sarcomeres in a myofibril and it further functions as a mechanical anchor to non-sarcomeric proteins of the cytoskeleton as well as the ECM. In this regard, it is key to absorbing and transmitting the energy from the above-described filament sliding. However, the Z-disc does not only hold function as a physical anchor, but it is also thought to serve as a mediator, transducer and transmitter of several biomechanical and biochemical signaling cascades (Pyle and Solaro 2004). Arguably the major protein

[1] In the literature, there is no strict consensus regarding the nomenclature of the terms *Z-disc* and *Z-line*. In this thesis, we typically refer to Z-discs, when considering the functional complex of proteins, whereas we refer to Z-lines, when considering the morphological distinct lines that demarcate each sarcomere at both ends.

of the roughly 100 nm wide Z-disc is the 2-isoform of α-actinin (α-actinin 2; sarcomeric α-actinin; ACTN2), which, together with several few other proteins, cross-links sarcomeric actin and reverses its polarity. ACTN2 is part of the spectrin superfamily and forms anti-parallel homodimers that are 200 kDa in size (Ribeiro et al. 2014). It consists of an N-terminal actin-binding domain, a long rod with spectrin-like repeats, and a C-terminal calmodulin-like domain with two pairs of EF hands, where the latter are involved in the binding of the bidirectional spring-like protein titin. Notably, ACTN2's actin- and titin-binding activity appear to be dynamically regulated, and, unlike in non-muscle actinin isoforms, the binding of actin to sarcomeric actinin is Ca^{2+}-insensitive (Ribeiro et al. 2014). Despite its crucial role in the Z-disc, ACTN2 accounts for a mere 20% of the total Z-disc protein content (Pyle and Solaro 2004).

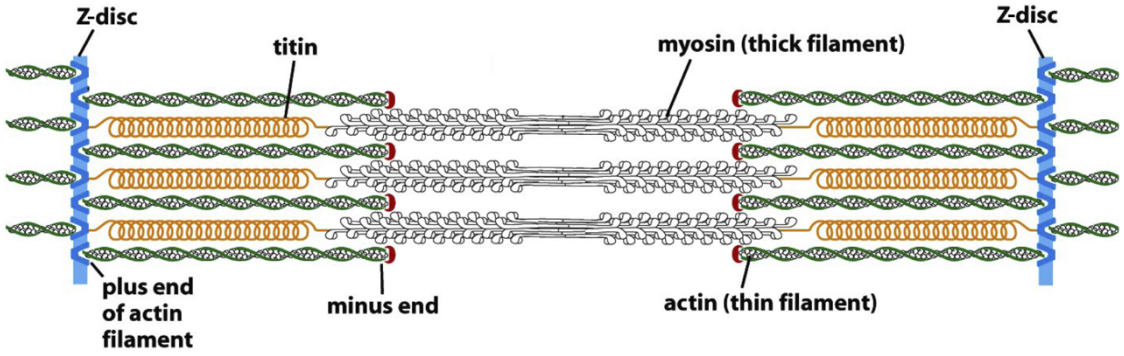

Figure 1-1: Sarcomere and Z-disc. Schematic depicting the Z-disc and the major sarcomeric proteins which are relevant in the context of this thesis. See text for details. Modified from Alberts et al. (2008), with permission of the Garland Science publishing group.

1.4 The Contractile Cycle of Cardiomyocytes

Excitation-contraction coupling is essentially the process, during which an electrical signal leads to mechanical contractions of cardiomyocytes (Bers 2002). Part of this process is the acto-myosin cross-bridge cycle, during which chemical energy is converted to mechanical energy via myosin power strokes, which, in turn, lead to the above-described sliding of thick filaments past thin filaments and thus contractions of sarcomeres. The contractile cycle in cardiomyocytes is being initiated via a series of ion fluxes, called action-potential. In brief, an initial membrane depolarization via sodium channels leads to a calcium influx via L-type Ca^{2+}-channels. This Ca^{2+}-influx then triggers a further Ca^{2+}-release from the sarcoplasmic reticulum into the cytosol. This process, termed calcium-induced calcium release, increases the cytosolic calcium concentration by roughly two orders of magnitude, resulting in a binding of Ca^{2+} to the troponin complex, which, in turn, induces a conformational change in the helix protein tropomyosin, thereby exposing previously concealed myosin-binding sites on actin.

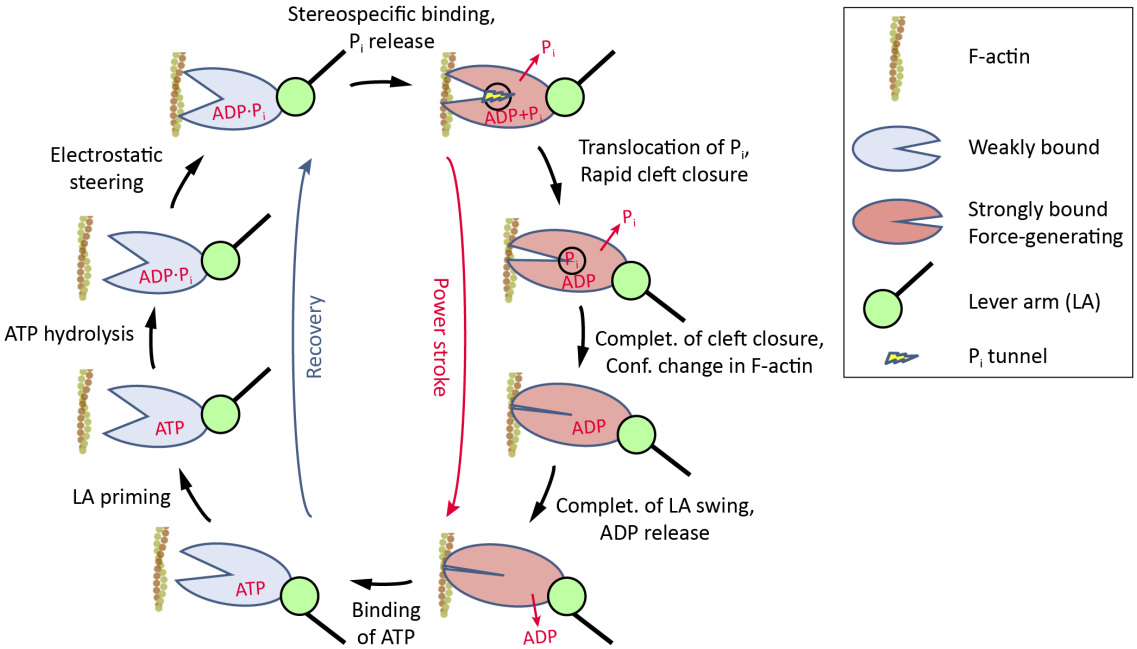

Figure 1-2: The acto-myosin cross-bridge cycle. Schematic depicting several putative steps of the ATPase cycle of myosin as performed on actin during cardiac contractions. See text for details. Modified from Houdusse and Sweeney (2016), with permission of *Elsevier*.

This Ca^{2+}-induced exposure of myosin-binding sites enables for the above-mentioned acto-myosin cross-bridge cycle (Figure 1-2). In this motor cycle, chemical energy from the hydrolysis of ATP is harvested and transduced to mechanical energy as myosin performs power strokes on actin. The detailed mechanisms of this chemomechanical cycle remain controversial, however, according to the most recent models, it is thought to comprise at least four strongly actin-bound states as the force-generating steps and four recovery states with weak affinity of myosin for actin (Houdusse and Sweeney 2016): At the start of the cycle, which is also the end of the power stroke, the nucleotide-free myosin head is strongly bound to actin in a rigor state. This state is relatively short-lived, as binding of ATP leads to detachment from the actin filament into the post-rigor state, which is the start of the recovery stroke. During this recovery stroke, first the myosin lever arm is being reprimed and the motor ends up in a pre-power stroke state. In this pre-power stroke state, ATP hydrolysis occurs with the products, ADP and inorganic phosphate (P_i), being tightly bound to the protein. Next, the myosin heads are being oriented on the actin filaments via a non-specific, electrostatic steering, resulting in weak attachment. The myosin heads then transition to a strong, stereospecific binding to actin, ending up in the P_i-release state, and force-generation is initiated. In this state, the actin-binding cleft is still open relatively wide, and P_i leaves the active site of myosin into a phosphate release-tunnel. This translocation of phosphate gates a first, rapid closure of the actin-binding cleft, which is coupled to a first, partial lever arm swing. Next, a transition to the strongly-bound ADP state occurs, which is

accompanied by the completion of the cleft closure and a conformational change in filamentous actin. Finally, the lever arm swing is completed and ADP is released.

1.5 Fluorescence Live Cell Microscopy

Over the past decades, fluorescence microscopy has become a – if not the – major imaging technique and analytical tool in the life sciences. Like other optical microscopy techniques, it takes advantage of the optical transparency of cells, and allows for non-invasive imaging of their interior in three dimensions. Owing to the intrinsic selectivity of fluorescent tagging, cellular constituents, such as proteins and nucleic acids, can be specifically localized and traced. It is this specific labeling that has made fluorescence microscopy so very useful to the life sciences. Aside from direct labeling, the standard method of choice is immunofluorescence, where the specificity of antibodies to their antigen is used to target fluorescent markers to the biomolecule of interest. One of the main drawbacks of this method is that it remains mainly limited to fixed (dead) cells when targeting molecules inside a cell. In this regard, the discovery of the green fluorescent protein (GFP) in 1962 (Shimomura et al. 1962) and the subsequent development of fluorescent proteins as fluorescent probes have revolutionized investigations of cellular processes in living organisms (Chudakov et al. 2010). Expression of fluorescent proteins is commonly considered to be innocuous to live cells, although deleterious effects, including induction of apoptosis or oxidative stress, have been reported (Ansari et al. 2016). Despite these potential limitations, fluorescent proteins display a unique and versatile probe, as they can be specifically fused to other proteins, i.e. cells can be genetically encoded to express certain proteins of interest as fusions to fluorescent proteins, enabling investigators to monitor cellular processes during live-cell imaging experiments.

1.6 Imaging of the Sarcomeric Network in Live Cardiomyocytes

Various groups have aimed to image sarcomere dynamics in live CM by tracing sarcomeric proteins and employing both, fluorescence microcopy or other lens-based microscopy techniques. Dabiri et al. (1997) transfected embryonic chicken cardiomyocytes with plasmids encoding for an ACTN2-GFP fusion protein, and were able to image myofibril formation in epifluorescence microscopy over time spans as longs as 28 hours at imaging intervals as short as two hours. In more dynamic studies, both, Hersch et al. (2013) and Shintani et al. (2014) transiently transfected rat CM with plasmids encoding for ACTN2-GFP fusion proteins to trace movements of Z-lines in real time. Hersch et al. (2013) transfected late embryonic rat CM, and imaged beating CM in epifluorescence at 17 fps. The authors then analyzed the displacement of Z-lines in single myofibrils to estimate contractile strain. Similarly, Shintani et al. (2014) transfected neonatal rat CM, and imaged contracting CM in epifluorescence at up to 50 fps. Here, the authors tracked the displacement of several few Z-lines to deduce relevant parameters of sarcomeric beating, such as contraction amplitude

and shortening/relaxation velocity. Ribeiro AJS et al. (2015) chose to label sarcomeres in hPSC-CM by transfecting cells with the commonly used F-actin marker Lifeact®. They then analyzed epifluorescence time lapse images at 5-10 fps to deduce average sarcomere contraction amplitudes, and, by employing particle tracking velocimetry, traced overall myofibrillar movement. In a unique approach, Serizawa et al. (2011) conjugated quantum dots to ACTN2 antibodies and applied the conjugates to stripped adult rat CM treated with lipid reagent FuGENE HD®. The authors were then able to extract the lengths of several individual sarcomeres from epifluorescence images at up to 39 fps.

Although the above-described studies all display remarkable approaches to the studying of the dynamics of fluorescently labeled sarcomeres, none of the presented fluorescence marker systems allowed for high-throughput or longitudinal studies. These drawbacks mainly arise from general concerns associated with cell transfections. One major concern, particularly with transient transfections, is, that in these transfected cells, the fluorescent fusion proteins are typically being overexpressed. Such overexpression of proteins can, inter alia, result in mislocalization or aggregation of proteins, violate gene dosage, affect protein folding, or simply lead to non-uniform fluorescent labeling (Gibson et al. 2013; Ratz et al. 2015). Further artefacts result from the interference of the transfection process itself with the viability of cells, or, in particular, with the contractility of cardiomyocytes. Moreover, non-proliferative cardiomyocytes are generally reported to be difficult to transfect, which holds particularly true for adult CM, such that researchers have often turned to transduction, i.e. the introduction of foreign DNA via viral vectors. In transduction of CM, infection rates of up to 100% can be achieved via adenoviruses or via adeno-associated viruses, however, expression time is either short (roughly two weeks, adenoviruses) or transferred gene size is relatively small (~ 5 kB, adeno-associated viruses; Louch et al. 2011). Accordingly, these drawbacks have motivated approaches to visualize sarcomeres in live CM that do not rely on cell transfection or transduction. One of these approaches exploits endogenous fluorescent labeling of (sarcomeric) proteins, which is the strategy of choice in this project, and which will be discussed in Section 1.8.

Non-fluorescence-based strategies mainly employ either standard bright-field microscopy or second harmonic generation microscopy. Standard bright-field imaging remains a mainstay to many research applications, as it is almost entirely noninvasive and as it demands relatively inexpensive microscopy setups as well as relatively simple sample preparation. However, when it comes to investigating sarcomere dynamics at the subcellular level, it remains limited to the analysis of average sarcomere contractions as opposed to intersarcomeric dynamics, even with the latest advances in computer-based image analysis (Pasqualin et al. 2016). Employing second harmonic generation microscopy, Gao and colleagues were able to image details of myofibrillogenesis in live neonatal rat CM at high spatial resolution and high contrast (Liu et al. 2011; Yang et al. 2016). Although displaying another exceptional and noninvasive imaging technique, image acquisition speed was, however, too slow (4 s/frame) for an application in the studying of contractile dynamics.

All in all, this brief overview of previous approaches to the imaging of sarcomere dynamics in real time motivates the need for a stable sarcomere tracing system in live CM that allows for high contrast, noninvasive imaging of sarcomeres at high spatial and temporal resolution.

1.7 Genome Editing Using the CRISPR/Cas9 System

During the past decade, several programmable nuclease-based genome editing techniques have emerged as versatile methods for the targeted modification of the genome of, in principle, any cell type in any eukaryotic organism. In these techniques, nucleases can be directed to induce specific double-strand breaks (DSB) in the genomic DNA, and briefly, intrinsic cellular repair mechanisms can then be exploited to repair custom-defined DNA sequences into the specific genomic site (Figure 1-3-A). DSB are typically repaired via either non-homologous end joining (NHEJ) or homology-directed repair (HDR; Hsu et al. 2014). In NHEJ, broken ends are rejoined on the basis of short microhomologies. This mechanism is highly error-prone and often results in the introduction of small indels, that in turn can lead to frame-shift mutations or gene knock-outs (Hsu et al. 2014). In the much more accurate HDR, intact DNA templates (typically from the sister chromatids) are copied via homologous recombination to precisely repair DSB (Hsu et al. 2014). By inserting custom-designed donor DNA templates into the cells, HDR-mediated repair can thus be exploited to generate cell lines or even entire eukaryotic organisms with defined point mutations, gene knock-outs, or gene knock-ins (Hsu et al. 2014).

Perhaps the most promising and currently the most rapidly developing of the genome editing techniques is the type II clustered regularly interspaced short palindromic repeats (CRISPR)/Cas9 system, which, in 2013, was first demonstrated to be employed for targeted gene modification in mammalian cells (Cong et al. 2013; Mali et al. 2013). Unlike in other nuclease-based genome editing techniques, such as in transcription activator-like effector nucleases (TALENs) or zinc finger nucleases (ZFNs), which recognize specific DNA sequences through direct protein-DNA interaction, the Cas9 nuclease is RNA-guided via Watson-Crick base pairing to induce double-strand breaks at specific sites of the target DNA. CRISPR systems display an adaptable immune mechanism in bacteria and archaea, which helps these prokaryotes to protect themselves against invading nucleic acids, e.g. bacteriophages or viruses (Sander and Joung 2014). In type II CRISPR, sequences from the invading DNA are integrated into arrays within the CRISPR locus of the prokaryote's genome, where protospacer sequences transcribed from the foreign DNA are segregated by direct repeats (Hsu et al. 2014; Sander and Joung 2014). Transcripts from these arrays are then processed into so-called CRISPR RNAs (crRNAs), which comprise sets of the segregated protospacers adjacent to a CRISPR repeat (Figure 1-3-B). A second, associated trans-activating CRISPR RNA (tracrRNA) hybridizes with the direct repeats of the crRNA to form an RNA duplex (guide RNA, gRNA), which complexes with the Cas9 nuclease. The

roughly 20 nucleotides of the crRNA, which correspond to the protospacer sequence, ultimately guide the Cas9 via Watson-Crick base pairing to induce a DSB in the target DNA if and only if the complementary sequence is adjacent to a short recognition sequence, known as protospacer adjacent motifs (PAMs). These PAMs are not present within the CRISPR locus, such that self-cleavage is intrinsically prohibited.

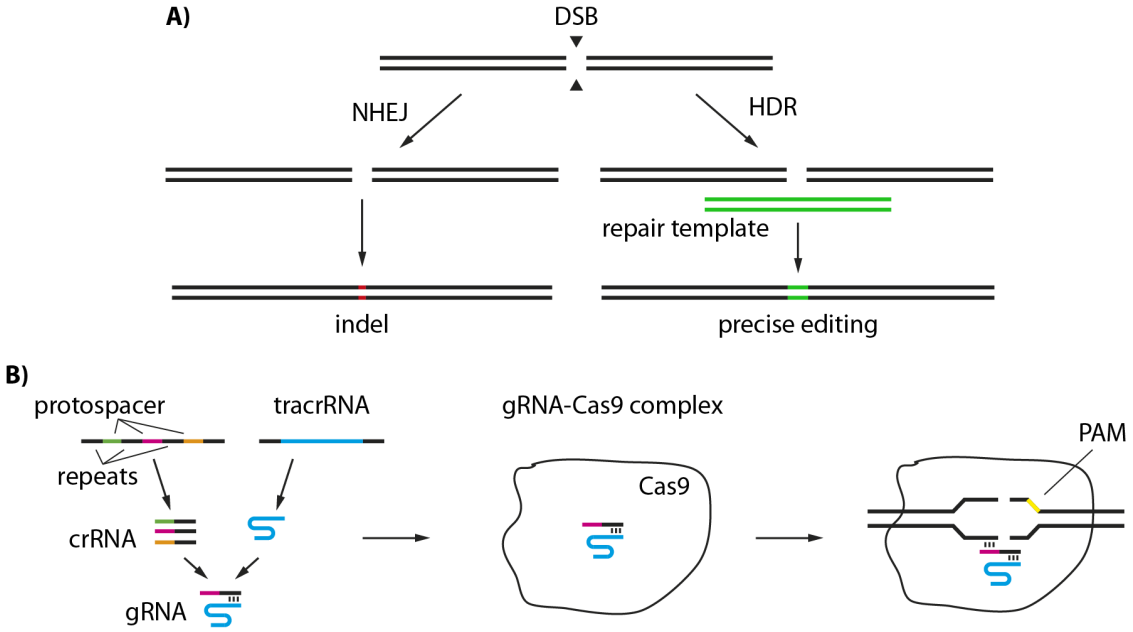

Figure 1-3: Rationale of CRISPR/Cas9-mediated genome editing. (A) Typical cellular repair mechanisms. DSB are repaired via either NHEJ or HDR. In the error-prone NHEJ, broken ends are rejoined on the basis of microhomologies. Here, insertion of variously sized indels are likely to occur. In HDR, DSB are repaired via homologous recombination based on the copying of repair templates. Homologous recombination is by far more accurate, thus allowing for precise genome editing. **(B)** Basic components of the type II CRISPR/Cas9 system. Arrays of protospacers transcribed from the invading DNA are segregated by direct repeats in the CRISPR locus of the prokaryote's genome. These arrays are processed into crRNAs, which hybridize to a second, associated tracrRNA to form guide RNA duplexes. The gRNA can then guide the Cas9 nuclease to induce DSB at specific sites in the target genome. The specific sites are dictated via Watson-Crick base pairing between the protospacers sequences of the crRNA and the complementary nucleotides in the target DNA, however, cleavage by Cas9 only occurs if this this complementary sequence is adjacent to a Cas9-recognized PAM. See text for details.

In the type II CRISPR/Cas9 system adapted from Streptococcus pyogenes (Sp), the PAM sequence lies directly at the 3'-end of the target sequence and is typically of form 5'-NGG (Hsu et al. 2014). To employ the Sp CRISPR/Cas9 system for HDR-directed genome engineering, three components thus have to be designed and introduced into the cell/organism of interest: *i)* a gRNA with the protospacer sequence of interest, *ii)* the Cas9 nuclease, and *iii)* a DNA donor template that comprises the custom-designed read-off se-

quence to be integrated into the host genome. The gRNA is typically being introduced as a chimeric single guide RNA, and should be designed to be as specific to the target site as possible, such that the likelihood for off-target cleavage is minimized. For Cas9 delivery, several different strategies have been employed (Wang H et al. 2016). Perhaps the simplest is to use transient transfections to introduce plasmids encoding for both, the Cas9 and the gRNA. An alternative, more elegant way is to utilize Cas9 ribonucleoproteins instead of Cas9 plasmids to avoid an exhausting induction of double-strand breaks and therefore minimize the likelihood for NHEJ (Wang H et al. 2016). The donor DNA is typically being introduced as single-stranded DNA, due to its lower cell toxicity as compared to double-stranded DNA. Importantly, the donor DNA needs to comprise matching homology arms, however, small silent point mutations need to be introduced to inhibit PAM sequences and thus cleavage of the donor strand.

All in all, the development of the CRISPR/Cas9 system as a genome editing tool has tremendously expedited our ability to engineer cell lines or organisms with defined custom-introduced or even corrected mutations. However, several challenges in terms of control and efficiency remain yet to be tackled (Wang H et al. 2016). Two important concerns in the context of this thesis are *i)* the possibility of off-target effects, and *ii)* the occurrence of NHEJ instead or next to the desired HDR. Off-target effects may arise not only because the protospacer-plus-PAM sequence might be present elsewhere in the genome, but also because it has been demonstrated that Cas9 can tolerate up to five mismatches within the guide sequence (Hsu et al. 2014). Further, alternative Cas9-recognized PAM sequences, e.g. 5'-NAG, have been reported (Hsu et al. 2014), such that both, 5'-NGG and 5'-NAG need to be considered when evaluating potential off-target sites. The occurrence of imprecise NHEJ, and, in turn, insertion of indels, is a highly frequent concern in nuclease-based genome engineering, as in eukaryotic cells, NHEJ naturally occurs at by far higher rates as compared to HDR (Alberts et al. 2008). Accordingly, modulating the HDR:NHEJ ratio towards HDR is an important step when employing CRISPR/Cas9, and a number of experimental strategies have been developed to increase the ratio (Wang H et al. 2016).

1.8 CRISPR/Cas9-mediated Endogenous Fluorescent Labeling

One of many applications of the CRISPR/Cas9 system, or genome engineering in general, is endogenous fluorescent labeling of cellular component such as proteins. Unlike in plasmid-driven overexpression of target-protein-fluorescent-protein-constructs (see Section 1.6), the sequence encoding for a fluorescent protein can here be integrated into the specific genomic locus of a target protein, such that the resulting fluorescent fusion protein is being expressed endogenously under the target protein's native promotor, and thus at close-to-native expression levels (Ratz et al. 2015). Such endogenous fluorescent tagging, in principle, overcomes general concerns of transfection- or overexpression-based fluorescent labeling (Section 1.6), and should allow for live-cell imaging as well as functional studies in

longitudinal experiments. If applied to pluripotent stem cell lines such as hPSC, an unlimited number of fluorescently labeled differentiated cells can be derived from the knock-in stem-cell line, which is particularly intriguing, when aiming for fluorescent labeling in non-proliferative cell types, such as cardiomyocytes or neurons.

Since the first demonstration of CRISPR/Cas9 as a genome editing tool in 2013, a number of articles have been published presenting successful endogenous fluorescent protein labeling in human cell lines. Ratz et al. (2015) first reported endogenous protein tagging for fluorescent live-cell imaging, when they tagged three proteins (vimentin, zyxin, and high mobility group AT-hook 1) in human U2OS cells with the fluorescent protein rsEGFP2. In total, they generated three homozygous and 27 heterozygous knock-in cell lines. In all heterozygous lines, insertion of 1 to 11 base pair indels was observed in the untargeted allele, which resulted in either a frame-shift mutation or no mutated protein, depending on whether the indel occurred in the coding region, or in the untranslated region. Despite these indels, the authors showed that typical overexpression-induced artefacts were avoided via endogenous labeling, and were able to perform sub-diffraction resolution imaging on the knock-in cell lines. Roberts et al. (2017) generated 10 hiPSC lines with different GFP-labeled proteins, including beta-actin and non-muscle myosin heavy chain IIa. Notably, the authors observed robust expression of pluripotency markers, suggesting that the level of differentiation of these stem-cell lines remained unaltered during the knock-in process. In 2018, Sharma et al. (2018) published first results from a study which was conducted simultaneously to this thesis project, and which aimed to fluorescently label sarcomeres in hPSC-CM. The authors here demonstrated endogenous fluorescent tagging of titin in functional hiPSC-derived cardiomyocytes, and showed proof-of-principle epifluorescent imaging of beating fluorescent CM at up to 40 fps for dynamic analysis of contractility.

Most recently and during the write-up of this thesis, Passier and colleagues published a mRubyII-ACTN2 and GFP-NKX2.5 double reporter line (Ribeiro et al. 2020), which the authors generated via CRISPR/Cas9-mediated genome editing in a previously characterized hESC line comprising a GFP-NKX2.5 transgene. Similar to our approach, the authors then derived CM with a fluorescent ACTN2 tag and investigated the contractile response to substrate stiffness.

1.9 Aims of this Thesis

The aim of this thesis project was to develop a stable fluorescence indicator system which allows for the robust *in-vitro* characterization of sarcomere contractility of hPSC-CM in live-cell microscopy. The novel indicator system was designed to label one of the sarcomeric proteins, and as such to allow for longitudinal studies which should a) help in understanding basic mechanisms of sarcomere dynamics and synchronization of hPSC-CM, and b) prove useful to the potential use of hPSC-CM in high-throughput drug screening assays.

We hypothesized that

A) The sarcomeric network in functional hPSC-CM can be fluorescently tagged via endogenous labeling
B) Endogenous fluorescent labeling allows for robust longitudinal studies in high-speed, live-cell fluorescence microscopy
C) Micropatterning of fluorescently labeled hPSC-CM facilitates high-throughput analysis of sarcomere contractility under defined experimental conditions
D) Computer-based automated analysis of sarcomere contractility allows for the extraction of mechanistic insight from the morphological feedback of fluorescently labeled sarcomeres

We further sought to prove applicability of the fluorescence indicator system to the analysis of how ECM rigidity influences sarcomere contractions and to the evaluation of drug interference on CM contractility.

2 Materials and Methods

2.1 Cells and Cell Culture

2.1.1 Human Embryonic Stem Cells

The import and experimentation with human embryonic stem cells was approved by the Central Ethics Committee for Stem Cell Research (ZES, permit #12, AZ 1710-79-1-4-16). Human embryonic stem cells, line 2 (HES-2) were obtained from Embryonic Stem Cell International (Singapore) and initially adapted to cell culture in hES medium (Section 6.3) on γ-irradiated human foreskin fibroblasts (SCRC-1041, American Type Culture Collection, Manassas, USA) as feeder cells[2]. After feeder removal, cells were plated to T-75 flasks coated with growth factor-reduced Matrigel™ (Cat 354230, BD Biosciences, Franklin Lakes, USA) and cultured in E8 medium (Section 6.3) with daily media changes at 37 °C and 5% CO_2. Passaging of HES-2 was conducted every two to three days at roughly 80% confluency in 0.5 mM ethylenediaminetetraacetic acid (EDTA) solution (Section 6.4).

2.1.2 Transgenic HES-2-derived Cardiomyocytes

After CRISPR/Cas9-mediated genome editing for an ACTN2-Citrine fusion protein and differentiation of the transgenic HES-2 lines (HES-2-ACTN2-Citr) to CM (Section 2.2), HES-2-ACTN2-Citr-derived CM (HES-2-ACTN2-Citr-CM) were cultured in Matrigel™-coated T-25 or T-75 flasks in serum-free basal medium (RPMI basal medium, Section 6.3) at 37 °C and 5% CO_2. After differentiation, HES-2-ACTN2-Citr-CM were cultured at least five days prior to preparation for experiments, while the culture medium was changed every other day. For experiments, CM monolayers were single-cell dissociated (Section 6.4), and plated to the experimental culture dish of interest (Section 2.3). Single-cell dissociation was conducted at least three days prior to experiments. Contractility experiments (Section 3.3) were run in either, RPMI basal medium at 0.4 mM Ca^{2+}, or in Dulbecco's modified Eagle's medium (DMEM, Section 6.3) at 0.8 mM Ca^{2+}, as noted in the respective Results sections. If experiments were conducted in DMEM, culture medium was changed from RPMI to DMEM after plating to the experimental culture dish.

[2] HES-2 maintenance on feeder cells and feeder removal was pursued by Mrs. Daria Reher, Iris Quentin, and Krasimira Sharkova, Institute of Pharmacology and Toxicology, University Medical Center Goettingen.

2.2 CRISPR/Cas9-mediated Endogenous Tagging of Sarcomeric Alpha-Actinin

To visualize Z-bands in live hESC-derived CM, we aimed to endogenously label sarcomeric actinin in HES-2 with the Citrine variant of the yellow fluorescent protein (YFP) via the CRISPR/Cas9 system. We chose Citrine as our fluorescent marker protein, as it exhibits strong, photostable fluorescence (Griesbeck et al. 2001), and as it is well-suited for our custom-built 592-nm stimulated emission depletion microscopy system.

2.2.1 Design of CRISPRs and Donor Vectors

Intending to tag the C-Terminus of ACTN2, we aimed to integrate the Citrine coding sequence at the 3'-end of the last exon (exon 21) in the ACTN2 gene sequence (Figure 2-1-A). To this end, we i) obtained two gRNAs (Sigma Aldrich Chemie GmbH, Taufkirchen, Germany) with 19 bp *binding domains* adjacent to PAM sequences within close proximity of the exon-21 STOP codon (Figure 2-1-B) and Table 2-1), and ii) designed three matching donor DNA strands, such that the gRNA/donor pairs would facilitate insertion of the Citrine coding sequence via HDR.

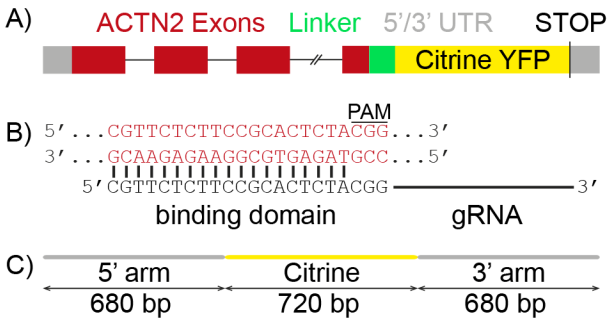

Figure 2-1: Design of gRNAs and donor vectors. (A) Intended position of the Citrine coding sequence in the genome after editing. Citrine was inserted immediately before the endogenous stop codon of the last exon in the ACTN2 gene sequence. **(B)** Schematic illustrating the base pairing between the gRNA's binding domain (here: gRNA-2) and the genomic DNA. **(C)** Schematic of the donor vectors. The donor vectors were comprised of the Citrine coding sequence flanked by left and right homology arms.

In detail, the donor vectors were comprised of the Citrine coding sequence (717 bp) flanked by left and right homology arms of roughly 700 bp in lengths (see Figure 2-1-C). One donor strand, termed *SerGG*, was designed to match gRNA-1. Here, we introduced the silent mutation TCC → TCT (encoding for the amino acid Serine) in the left homology arm, to inhibit gRNA-1's PAM sequence and thus cleavage of the donor strand (Table 2-2). In the case of gRNA-2, the PAM sequence CGG was part of the codon GGG encoding for Glycine. As all existing codons encoding for Glycine are of type GGn (Alberts et al.

2008), the PAM sequence could not be inhibited directly, and we thus introduced four[3] silent mutations (TCC → TCT, CTC → TTG, and GCA → GCT) to inhibit base pairing between gRNA-2's binding domain and the donor strand (see Table 2-2). This second donor strand we termed *4Mut*. Last, we designed a third donor strand termed *Linker*, in which we introduced an extra sequence encoding for the short linker peptide $(Gly)_5$-Ala between the left homology arm and the Citrine coding sequence. This flexible linker peptide was inserted to minimize potential folding perturbation between Citrine and ACTN2 (Doyle and Botstein 1996). The donor strand *Linker* comprised all above silent mutations to work with both gRNA-1 and gRNA-2.

Table 2-1: Binding domain sequences of gRNAs.

gRNA Name	Sequence (5' to 3')
gRNA-1	CCGCACTCTACGGGGAGAGCGA
gRNA-2	CGTTCTCTTCCGCACTCTACGG

Red writing denotes PAM sequences.

Table 2-2: Endogenous DNA sequence and sequences of donor vectors.

Donor name	Sequence
Endogenous	(…) -TCC-GCA-CTC-TAC-GGG-GAG-AGC-GAT-CTG-TGA
SerGG	(…) -TCT-GCA-CTC-TAC-GGG-GAG-AGC-GAT-CTG-CITRINE-TGA
4Mut	(…) -TCT-GCT-TTG-TAT-GGG-GAG-AGC-GAT-CTG-CITRINE-TGA
Linker	(…) -TCT-GCT-TTG-TAT-GGG-GAG-AGC-GAT-CTG-GGA-GGT-GGA-GGT-GGA-GCT-CITRINE-TGA

Sequences are only shown in the region of the near left homology arm. Pink highlighting denotes silent mutations, which were introduced to inhibit cleavage of the donor strands. Red writing marks the PAM sequences of the respective gRNAs. Highlighted in red is the endogenous STOP codon. CITRINE denotes the coding sequence for Citrine YFP, and highlighted in green is the sequence encoding for the small linker peptide $(Gly)_5$-Ala, which was inserted in the case of the donor termed *Linker*.

We purchased the donor DNA strands cloned into pUC57-Kanamycin plasmids (GENEWIZ UK, Takeley, UK), and in order to obtain linear DNA donor vectors, we

[3] We here also introduced a 5th silent mutation, namely TAC → TAT, where the C is part of the PAM sequence. However, as the PAM sequence is not part of the actual binding domain of the gRNA, this might not have had an effect on the base pairing between gRNA-2 and the donor strand.

amplified the insert via PCR using custom-designed primers (Table 6-1, Sigma Aldrich Chemie GmbH, Taufkirchen, Germany). The full sequences of the complete linearized donor strands are listed in Table 6-3.

The chimeric gRNAs were provided in ready-to-use *CRISPR plasmids* (Sigma Aldrich Chemie GmbH, Taufkirchen, Germany), which encoded both the respective gRNA (U6 promotor) and the Cas9 nuclease (CMV promotor). Moreover, a green fluorescent protein (GFP) was fused to the C-terminus of the Cas9, such that in the CRISPR workflow, Cas9-expressing cells could be selected via FACS (Section 2.2.2).

2.2.2 Integration of Citrine YFP into the ACTN2 Gene Locus

To integrate Citrine[4] into the ACTN2 Gene Locus, we co-transfected wild-type HES-2 with the gRNA/Cas9 plasmids and accordant donor DNA. Transfection was pursued via electroporation with cells in solution.

To prepare HES-2 for electroporation, cells were cultured in E8 medium in Matrigel™-coated T-75 culture flasks and passaged three days prior to transfection (Section 2.1.1). For electroporation, HES-2 were digested using 0.5 mM EDTA and were resuspended in E8 medium with 10 μM ROCK inhibitor. Cells were then counted, centrifuged (4 min at 200 x g), and for each CRISPR/donor pair, 2×10^6 cells were carefully resuspended in 100 μL of Amaxa™ buffer (Lonza, Basel, Switzerland). 3 μg of gRNA/Cas9 plasmid and 3 μg of linearized donor DNA were mixed into the cell suspension, and the samples were transferred into Amaxa™ electroporation cuvettes (Lonza, Basel, Switzerland). Cells were then electroporated using program A-23 of the Amaxa™ Nucleofector™ (Lonza, Basel, Switzerland). Immediately after electroporation, 500 μl of E8 medium with 10 μM ROCK inhibitor were added to the suspension and the samples were transferred to Matrigel™-coated T-25 culture flasks with E8 medium containing 10 μM ROCK inhibitor equilibrated at 37 °C. After 24 hours, E8 medium was changed, now containing no ROCK inhibitor. 48 hours after transfection, GFP-positive cells were FACS sorted[5] into Matrigel™-coated 96-well tissue culture plates containing E8 medium with 10 μM ROCK inhibitor to obtain single cell clones. As Cas9 enzymes were tagged with a GFP marker, transient exhibition of GFP signal indicated the expression of the gRNA/Cas9 plasmids within the HES-2. 24 hours after FACS sorting, E8 medium was changed and clones were cultured onwards with daily media changes at 37 °C and 5% CO_2.

One week after FACS sorting, clones were briefly examined using a binocular to rule out obvious polyclonality, and 12 to 14 days after FACS sorting, clones were re-seeded into two 96-well plates for each CRISPR/donor pair, respectively. Cells in one 96-well plate

[4] Herein and after, "integration of Citrine", in fact, refers to the integration of the DNA sequence encoding for Citrine into the ACTN2 gene locus, as described in Section 2.2.1.

[5] FACS sorting was conducted at the Cell Sorting Facility of the University Medical Center Goettingen with help of Mrs. Sabrina Becker.

were continued to be cultured, whereas cells in the second 96-well plate were used for PCR genotyping (Section 2.2.3). Citrine-positive clones were cultured further and passaged to increasingly larger area culture dishes. Clones were finally used for differentiation into cardiomyocytes (Section 2.2.4) or frozen to -80 °C for later use. Figure 2-2 illustrates the overall workflow of the genome editing process.

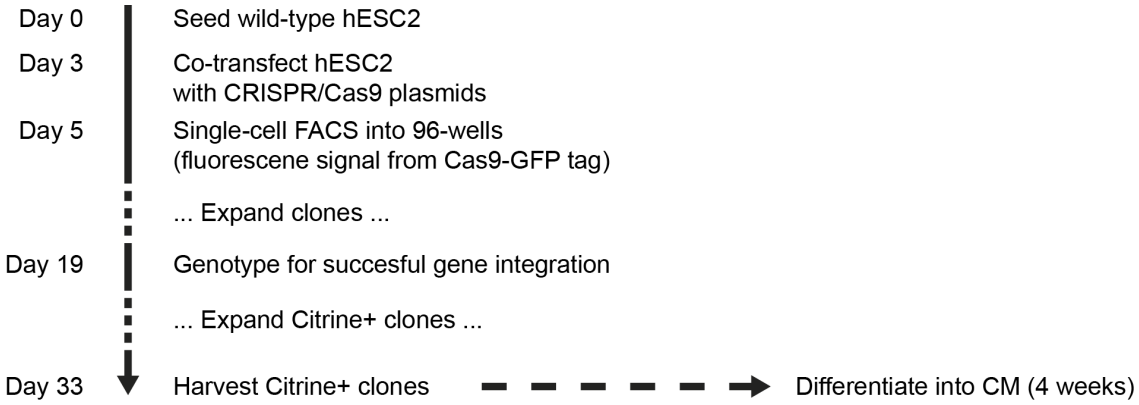

Figure 2-2: Workflow of the genome editing process using CRISPR/Cas9. Briefly, wild-type HES-2 were seeded and co-transfected with CRISPR/Cas9 and donor plasmids. Cells positive for a Cas9-GFP tag were single-cell sorted into 96-wells. The clones were then genotyped for successful gene integration, and Citrine-positive clones were eventually harvested and differentiated into cardiomyocytes.

2.2.3 Genotyping of CRISPR/Cas9-modified HES-2

As sarcomeric α-actinin is not being expressed in undifferentiated stem cells (Perán et al. 2010; Shen et al. 2016), we used PCR and gel electrophoresis for genotyping to test whether integration of Citrine had succeeded. For PCR, we custom designed forward and reverse primers binding roughly 500 bp before and behind the Cas9 cleaving sites, respectively (Table 6-2). For an integrated Citrine, PCR yielded amplifying products of roughly 1700 bp in lengths, whereas the unmodified fragment was roughly 1000 bp in length. Gel electrophoresis of the PCR products thus yielded a) solely a short (1000 bp) band for an unmodified genome, b) a short and a long (1700 bp) band in the case of successful but heterozygous integration of Citrine, and c) solely a long band for successful and homozygous modification (see Figure 2-3). PCR and gel electrophoresis protocols are listed in Section 6.4.

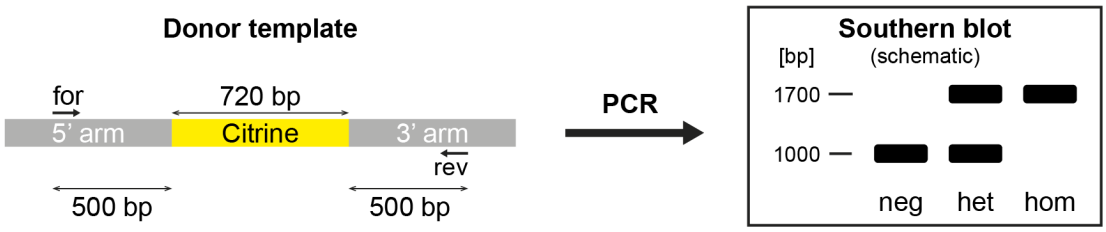

Figure 2-3: PCR genotyping to monitor successful gene integration. To test for successful integration of Citrine into the ACTN2 gene locus, we custom-designed forward and reverse primers binding roughly 500 bp before and behind the Cas9 cleaving sites (Table 6-2). PCR using these primers and subsequent gel electrophoresis thus yielded a) solely a short (1000 bp) band for an unmodified genome, b) a short and a long (1700 bp) band in the case of successful but heterozygous integration of Citrine, and c) solely a long band for successful and homozygous modification. Right-hand schematic represents no real data.

2.2.4 Cardiac Differentiation of HES-2-ACTN2-Citr

Cardiac differentiation of transgenic HES-2 lines (HES-2-ACTN2-Citr) was pursued in 2D and according to an optimized protocol (Hudson et al. 2012; Tiburcy et al. 2017)[6]. In brief, HES-2-ACTN2-Citr were plated at 2 x 10^3 cells/cm^2 in Matrigel®-coated T-25 flasks and cultured in E8 medium (see Section 6.3) for four days. Subsequently, the cells were cultured in mesodermal induction medium (Section 6.3) for three days, before being cultured in cardiac differentiation medium (Section 6.3) for another nine days. The cells were then cultured for a further five days in serum-free RPMI basal medium (Section 6.3). To eliminate non-myocytes via metabolic selection, the differentiated cells were finally cultured in selection medium (Section 6.3) for three days. Last, HES-2-ACTN2-Citr-CM were cultured in RPMI basal medium (Section 6.3) for recovery for another five days, before being prepared for downstream application.

2.3 Micropatterning of hPSC-derived Cardiomyocytes

To provide an anisotropic microenvironment for defined experimental conditions in potential high-throughput analysis of CM contractility, we aimed to micropattern hPSC-CM, or HES-2-ACTN2-Citr-CM in particular, to rectangular geometries of aspect ratio 7:1. To further investigate the influence of ECM rigidity on CM contractility, we employed micropatterning on substrates of two different stiffnesses. On the one hand, we micropatterned hPSC-CM on hard glass substrates, and, on the other hand, we plated CM on 15 kPa elastomer substrates to mimic the physiological microenvironment in the heart (Berry et al. 2006). Off-the-shelf substrates were purchased commercially (µ-dishes, Cat 81158 (glass)

[6] Cardiac differentiation was conducted by Krasimira Sharkova, Institute of Pharmacology and Toxicology, using the facilities of the Stem Cell Unit – Goettingen at the University Medical Center Goettingen.

and Cat 81391 (15 kPa), ibidi, Martinsried, Germany). Of the various ECM proteins available, we chose synthetic, animal-free vitronectin (Synthemax® II-SC substrate, Corning, Corning, USA). Throughout the project, three ECM proteins (laminin, fibronectin, and Synthemax®) were tested, and patterning of hPSC-CM on Synthemax® islands was most reliable, while it qualitatively also supported best contractile and morphologic properties (data not shown).

2.3.1 Design of Cell Patterns and Corresponding Photomasks

In the scope of this project, two separate photomasks with different cell patterns were designed. One photomask, named "Seven to One", was designed to provide rectangular stamp areas for isolated, single-cell cardiomyocytes of three different cell sizes at aspect ratio 7:1 (Figure 2-4-A). The second mask, named "Cell Talk", was designed to investigate the interaction between two cells. It provided stamp areas for pairs of cells, in which each two cells were seeded at different spacing (1 µm to 40 µm) and different respective geometries (parallel, in row, perpendicular; Figure 2-4-B). In the case of this second mask, all cells were of dimensions 70 µm x 10 µm. The spacing between pairs was 100 µm.

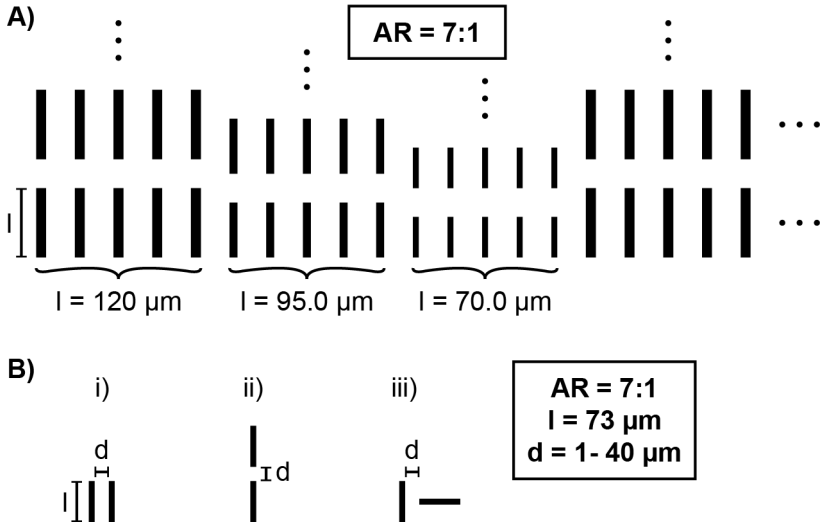

Figure 2-4: Design of cell patterns. Schematic depicting cell patterns for microcontact printing. **(A)** Patterns of the photomask named "Seven to One", which provided rectangular patterns for isolated, single-cell CM of three different cell sizes at aspect ratio 7:1. **(B)** Patterns of the photomask named "Cell Talk". Here, each two cells were seeded at different spacing (1 µm to 40 µm) and three different respective geometries, and all cells were of dimensions 70 µm x 10 µm. The spacing between pairs was 100 µm.

Drawings of the cell patterns were created using the software AutoCAD (Autodesk Inc., San Rafael, USA). The respective custom-designed soda-lime photomasks were purchased commercially (Compugraphics, Jena, Germany).

2.3.2 Microfabrication of Photoresist Masters via Soft Contact Lithography

Photoresist masters used as moldings for the production of polydimethylsiloxane (PDMS) stamps were fabricated by standard concepts of *soft lithography* inside a class 100 cleanroom (Xia and Whitesides 1998; Qin et al. 2010). Briefly, Czochralski silicon wafers (Microchemicals GmbH, Ulm Germany) were spin-coated with Series 3000 SU-8 negative epoxy-based photoresists (MicroChem, Newton, USA). Wafers were then illuminated through custom-designed soda-lime photomasks (Compugraphics, Jena, Germany), developed in photo developer to wash away non-illuminated regions of photoresist, and finally silanized before use for the fabrication of PDMS stamps (Figure 2-5).

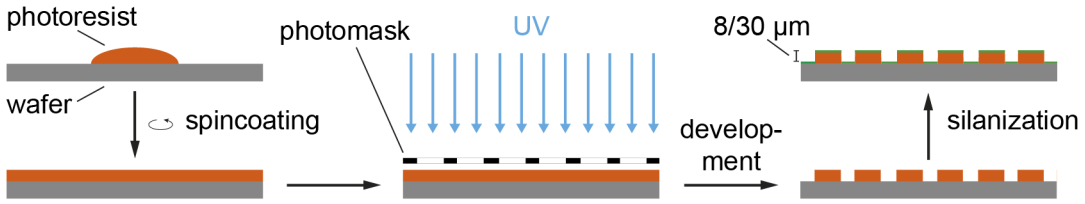

Figure 2-5: Fabrication of photoresist masters. Briefly, 8 or 30 µm thick layers of SU-8 negative photoresist were spin-coated onto silicon wafers and illuminated via custom-designed soda-lime photomasks. The wafers were then photo developed, resulting in microstructured photoresist patterns with cavities in the desired geometry.

Depending on the stiffness of the substrate to be micropatterned, we produced stamps and thus moldings of different depths. For stamping on hard glass substrates (Section 2.3.4), we aimed for 8 µm deep stamps and thus 8 µm deep moldings. To this end, we used type 3005 SU-8 negative photoresist (MicroChem, Newton, USA) and optimized our cleanroom protocol according to the Series 3000 SU-8 data sheet. First, wafers were cleaned in acetone in an ultrasonic bath for 10 minutes before being rinsed with isopropanol. After nitrogen-blowdrying, wafers were spin-coated with type 3005 SU-8. 2 ml of SU-8 were dispensed on the 2-inch wafers, and wafers were spun i) at 500 rpm for 10 seconds (ramp 100 rpm/s) and then ii) at 1800 rpm for 30 seconds (ramp 300 rpm/s), resulting in an 8 µm deep layer of photoresist. Wafers were soft-baked immediately afterwards on a 95 °C hotplate for five minutes. After cooling for 15 minutes, the wafers were then illuminated through the photomask for three seconds using a Karl Süss MJB4 mask aligner (Süss Microtech AG, Garching, Germany) operating in soft contact mode at a total power of 150 mJ/cm^2 at wavelengths 290 nm and 365 nm. Illumination was directly followed by post-exposure bake (PEB) at 65 °C for one minute and two minutes at 95 °C. Wafers were cooled for another 15 minutes and then developed by gently shaking them in AV-rev 600 photo developer (Microchemicals, Ulm, Germany) for two minutes and 40 seconds. After stopping the development in a gentle flow of isopropanol for 30 seconds, wafers were nitrogen-blowdried and hard baked on a 150 °C hotplate for 30 seconds.

Finally, the photoresist masters were silanized in order to prevent strong adhesion between the photoresist and the PDMS during stamp fabrication (see Section 2.3.3). To this end, the masters were placed inside a vacuum desiccator along with a small beaker containing a few drops of chlorotrimethylsilane (Sigma Aldrich Chemie GmbH, Taufkirchen, Germany). The vacuum pump was switched on for 30 minutes, and the masters were kept inside the vacuum overnight. The vacuum induced formation of silane vapors and ensured silanization of the photoresist. The next day, masters were baked in an oven at 70 °C to finalize silanization.

For stamping on soft elastomer substrates (Section 2.3.5), we aimed for stamps and thus moldings of 30 µm in depths. Here, we used type 3025 SU-8 negative photoresist (MicroChem, Newton, USA; ~ 2 ml per wafer), and the microfabrication process was similar to that for 8 µm moldings, however, with slight differences described as follows. Spin-coating was performed i) at 500 rpm for 10 seconds (ramp 100 rpm/s) and ii) at 3500 rpm for 30 seconds (ramp 300 rpm/s), resulting in a 30 µm deep layer of photoresist. Soft baking time was 15 minutes, exposure time was 12 seconds, and PEB time was one minute at 65 °C, followed by five minutes at 95 °C. Finally, as the standard development procedure did not yield sufficient results for these relatively deep cavities, we sonicated the wafers in developer for five minutes, followed by an extra five minutes of gentle shaking in developer. Hard bake time was one minute at 150 °C. Silanization with chlorotrimethylsilane was performed as with 8 µm deep masters.

2.3.3 Fabrication of PDMS Stamps

To produce PDMS stamps of 8 and 30 µm in depths, first PDMS and curing agent (Sylgard 184 kit, Dow Corning) were mixed at ratio 10:1 before being degassed in a vacuum desiccator for 20 minutes. Subsequently, roughly 5 mm thick layers of the PDMS mixture were cast onto the photoresist masters and the mixture was degassed for another 15 minutes. The PDMS was then cured at 60 °C for two hours. Finally, the cured PDMS was peeled off the photoresist masters and the micropatterned regions were cut to roughly 1 cm x 1 cm-sized stamps using a scalpel (Figure 2-6). Stamps were then sonicated in 70% ethanol for 15 minutes, air-blowdried, and stored in dust-free containers until further use.

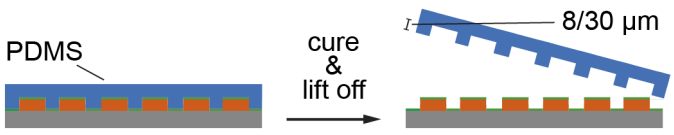

Figure 2-6: Fabrication of PDMS stamps. Briefly, a 10:1 mixture of PDMS and curing agent was cast onto the 8 µm or 30 µm deep photoresist masters, degassed and cured at 60 °C for two hours. The cured PDMS was then peeled off and the micropatterned regions were cut to roughly 1 cm^2-sized stamps using a scalpel.

2.3.4 Printing on Hard Substrates

For microcontact printing on hard glass substrates, we used 8 µm deep PDMS stamps (Section 2.3.3), and our protocol was similar to that described by Thery and Piel (2009). Stamps were incubated with 20 µg/ml of Synthemax® in PBS for 30 minutes at room temperature. After 30 minutes, the excess ECM protein solution was aspirated off, and the stamps were immediately washed once with PBS and twice with deionized water. Stamps were then air-blowdried. In the meanwhile, the glass substrates were plasma-oxidized in a table-top plasma cleaner (Harrick, Ithaca, USA) for 15 minutes in order to render the glass surface hydrophilic for increased protein binding. Immediately after plasma cleaning of the substrates, the stamps were inverted onto the substrates (Figure 2-7). For increased protein transfer, 50 g weights were placed on top of the stamps, and stamps were kept in contact with the substrates for 5-10 minutes. Stamps were then removed, and to block cell adhesion to unstamped regions, substrates were incubated with a 0.1 mg/ml poly(L-lysine)-g-poly(ethylene glycol) solution (PLL-g-PEG, SuSoS, Duebendorf, Switzerland) in 10 mM HEPES (pH 7.4) for 30 minutes at room temperature. The PLL-g-PEG-treated substrates were finally washed extensively with PBS and stored immersed in PBS at 4 °C until the seeding of cells.

Prior to cell deposition, PBS was exchanged for culture medium and substrates were equilibrated to 37 °C for 45 minutes. 50 to 150 x 10^3 cells were then added to the solution and substrates were placed inside a cell culture incubator at 37 °C and 5% CO_2. After 30 minutes, substrates were washed in a gentle flow of medium to remove unattached cells. Substrates were then placed back inside the incubator and attached cells were cultured until imaging.

Figure 2-7: PDMS stamping on glass substrates. 8 µm deep PDMS stamps were incubated with 20 µg/ml of Synthemax® for 30 minutes. The protein solution was then air-blowdried off, and the stamps were inverted onto the glass substrates, which had priorly been rendered hydrophilic using a plasma cleaner. Finally, the stamps were removed and the substrates were incubated with a 0.1 mg/ml solution of PLL-g-PEG to block unstamped regions.

2.3.5 Printing on Soft Substrates

For relatively soft substrates, like the 15 kPa elastomeres used in this project, conventional microcontact printing via direct PDMS stamping did not yield reproducible results. We thus turned to a method described by Hampe et al. (2014), in which the ECM proteins are

deposited through cavities in microstructured lift-off epoxy membranes (Hampe et al. 2014). Briefly, membranes were formed from a UV-curable epoxy resin which was being filled between a PDMS stamp and a glass slide, resulting in membranes which resembled the negative of the PDMS stamps. The membranes were then placed onto the substrate and the elastomer substrates were incubated with the protein solution through the microstructured cavities of the epoxy membranes (Figure 2-8).

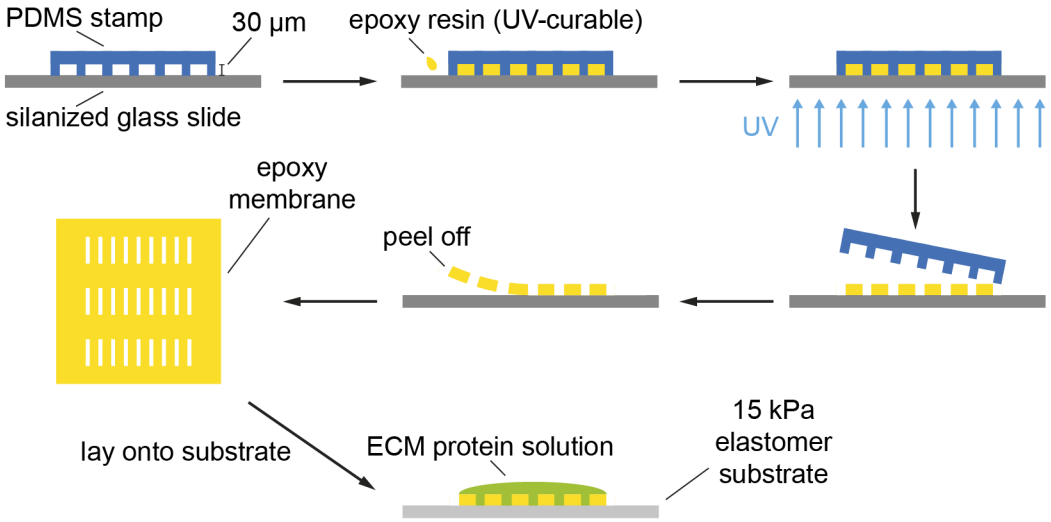

Figure 2-8: Microcontact printing on relatively soft 15 kPa substrates. Briefly, a 30 µm deep PDMS stamp was inverted onto a silanized glass slide, and a UV-curable epoxy resin was filled between the stamp and the glass. The resin was then cured and the stamp was lifted off the glass slide. Peeling off the cured resin off the glass yielded an epoxy membrane which resembled the negative of the PDMS stamp. The membrane was then placed onto the elastomer substrate and the substrate was incubated with the ECM protein solution (Synthemax®) through the cavities of the epoxy membrane.

For epoxy membrane preparation, we used 30 µm deep PDMS stamps, as described in Section 2.3.3. PDMS stamps, as well as glass slides, were silanized with trichloro(1H,1H,2H,2H-per-fluorooctyl)silane (Sigma Aldrich Chemie GmbH, Taufkirchen, Germany) for 45 minutes in a vacuum desiccator. This step minimized the adhesion between the epoxy resin and the PDMS stamp as well as the adhesion between the epoxy resin and the glass slide. The PDMS stamp with an additional 50 g weight was placed onto the glass slide, and uncured epoxy resin (EPO-TEK OG142-87, J. P. Kummer, Augsburg, Germany) was filled in between the stamp and the glass via capillary forces. The epoxy resin was then cured on a UV light transilluminator table (Intas, Goettingen, Germany) for 30 minutes. Finally, the stamp was lifted off and the hardened epoxy membrane was peeled off the glass slide and stored in a dust-free container until further use.

In order to form microstructured protein islands, the epoxy membranes were gently placed onto the elastomers and the sandwich was incubated with the ECM protein solution. Differently from Hampe et al. (2014), we did not degas the protein solution after deposition in

a vacuum, as this step resulted in strong adhesion between the epoxy membranes and the elastomer and thus destruction of the elastomer when lifting off the membranes. However, previously to membrane deposition, we instead placed the elastomer into a plasma cleaner for 15 seconds at low power level. This step increased adhesion between the proteins and the elastomer, and minimized coating issues resulting from air bubbles in the cavities. After membrane deposition, 20 μg/ml of Synthemax® in PBS were placed on top of the membrane, and the elastomer substrate was incubated with the solution for 60 minutes at room temperature. The excess protein solution was aspirated off and the culture dish was washed three times with PBS. The epoxy membrane was then peeled off and the substrate was incubated with a 0.1 mg/ml PLL-g-PEG solution in 10 mM HEPES (pH 7.4) for 30 minutes at room temperature, as described in Section 2.3.4. The PLL-g-PEG-treated substrates were finally washed extensively with PBS and stored immersed in PBS at 4 °C until the seeding of cells, as described in Section 2.3.4.

2.4 Fluorescence Microscopy

2.4.1 Resonant-scanning Confocal Microscopy

Fluorescence live-cell video imaging was performed using a commercial resonant-scanning confocal microscope (TCS SP5 X, Leica Microsystems, Wetzlar, Germany). The microscope system was equipped with a white-light laser source for fluorescence excitation, and a resonant scanner, that allowed for high-frequency confocal microscopy at 8 kHz line scanning. Unless noted, confocal images were recorded using a NA 1.4 63X immersion oil objective (HCX PL APO 63X / 1.40 - 0.60 Oil CS, Leica Microsystems, Wetzler, Germany) with the confocal pinhole set to 1 Airy. Citrine YFP was excited at 514 nm, and emitted fluorescence was collected via photo multiplier tubes, built into the TCS SP5 X system. To ensure optimal culture conditions during live-cell experiments, video images were recorded at 37 °C and 5% CO_2.

2.4.2 Immunofluorescence

In some characterization experiments, hPSC-derived CM were fixed and immunostained for α-actinin 2 and/or actin. Sarcomeric actinin was labeled using an anti-ACTN2 primary antibody from mouse (1:1000, Cat A7811, Sigma Aldrich, Taufkirchen, Germany), and either, anti-mouse-Alexa633 from goat (1:250, Cat A-21052, Thermo Fisher Scientific, Waltham, USA), or Oregon Green 488-conjugated anti-mouse from goat (1:250, Cat O-11033, Thermo Fisher Scientific, Waltham, USA) as secondary antibodies. Labeling of actin was conducted using phalloidin-Atto550 (1:250, Cat 19083, Sigma Aldrich, Taufkirchen, Germany). The detailed protocol for immunolabeling is listed in Section 6.4.

Fluorescence images of immunostained hPSC-CM were recorded with either, the above-described confocal microscope, or an inverted epifluorescent microscope (AxioVert, Zeiss, Oberkochen, Germany) equipped with a 10X air objective and a mercury arc lamp.

2.5 Video Image Analysis

To analyze confocal video images, each consisting of roughly 350-400 frames, we custom-developed an analysis algorithm, which yielded not only basic characteristic parameters of cardiomyocyte beating, but also allowed for the evaluation of the coherence of sarcomere contractions within single cardiac myofibrils. Scripting was performed in Python®, version 2.7 (Python Software Foundation; Python Language Reference, version 2.7; available at http://www.python.org). Large parts of the video analysis script were written by PhD candidate Daniel Haertter, M.Sc. (Drittes Physikalisches Institut, Georg-August-University Goettingen). Parts of the video analysis script, namely the peak finding algorithm and several minor adaptations, were re-written by the author of this thesis to meet the demands of the video data recruited in the scope of this thesis. A copy of the video analysis script can be found in the Appendix (Section 6.5). Meta-analysis Python® scripts, designed to summarize the results from single video analysis and to perform statistical analysis as well as data illustration, were written entirely by the author of this thesis. Analysis of all data shown in this thesis was performed by the author.

As the development of the analysis algorithm was part of this thesis project, the detailed algorithm is demonstrated in the Results section (Section 3.3.1). Notably, at the start of the analysis of each beating CM, the raw image time series were transferred to ImageJ-based Fiji, version 2.0 (Schindelin et al. 2012). Here, the intensity profiles along the myofibrils of interest over time were extracted and saved into data spreadsheets using the ImageJ/Fiji macro StackProfileData (StackProfileData.txt, available at https://imagej.nih.gov/ij/macros/). The data spreadsheets were then transferred to Python®, and the custom-written analysis algorithm was employed.

2.6 Statistics

Unless noted, data are presented as mean ± one standard deviation (STD). Statistical analysis was performed in Python® using Student's unpaired t-test when comparing two groups. In the case of concentration-response curves, we performed a repeated measures analysis of variance (rmANOVA) with post-hoc paired t-tests for pairwise comparisons. Normal distribution was assumed, values of $p < 0.05$ were considered statistically significant. In boxplot data illustrations, boxes represent 1st and 3rd quartiles, red lines represent the median, and whiskers represent minimal and maximal values observed.

3 Results

3.1 Micropatterning of hPSC-derived Cardiomyocytes

As described in Section 2.3, we aimed to employ the techniques of microcontact printing to force hPSC-CM to assume physiological aspect ratios of 7:1. In Section 3.1.1, we describe several crucial steps in implementing the technique and optimizing it to our specific needs. In Sections 3.1.2 and 3.1.3, we show the resulting cardiomyocyte patterns on glass and 15 kPa elastomer substrates, respectively.

Notably, the micropatterning process was implemented in our laboratories before the genome-edited knock-in lines were available, such that the micropatterning was established using hiPSC-CM as test cells (available in-house (Tiburcy et al. 2017), generated using CytoTune Sendai virus reprogramming kit, Thermo Fisher Scientific, Waltham, USA). Accordingly, microscope images of micropatterned CM in the following subsections display hiPSC-CM rather than HES-2-CM. Handling and micropatterning of hiPSC-CM was the same as for HES-2-CM (Tiburcy et al. 2017).

3.1.1 Fabrication of Photoresist Masters and PDMS Stamps

The first crucial step in implementing the micropatterning technique was the optimization of the soft lithography process such that it reproducibly yielded well-defined photoresist moldings for the fabrication of the PDMS stamps. In the case of 8 μm deep masters, the cleanroom process was relatively straightforward. Here, the main step that needed to be optimized was to adjust the exposure time to the thickness of the type-3005 SU-8 negative photoresist layer, and we narrowed it down to three seconds for our setup. Figures 6-1-A and 6-1-B show 8 μm deep masters which were exposed for three and eight seconds, respectively. The small structures on the wafer in Figure 6-1-B show clear signs of overexposure where the photoresist couldn't fully be washed out during the development process, whereas all structures in Figure 6-1-A were fully developed. Figures 3-1-A and 3-1-B show higher magnification images of final 8 μm deep masters illuminated for three seconds via photomasks "Seven to One" and "Cell Talk", respectively (Section 2.3.1). The images clearly show well-resolved structures, even in the case of the Cell Talk geometries, where the spacing of rectangles was only 1 μm (upper right and lower right of Figure 3-1-B).

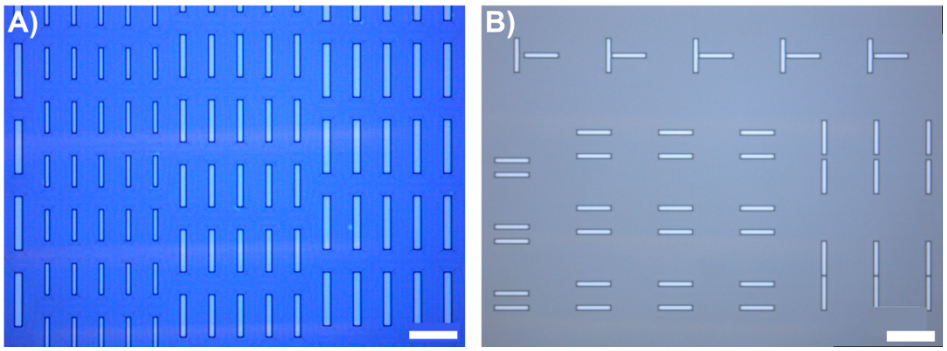

Figure 3-1: 8 µm deep photoresist masters. (A) Wafer illuminated through photomask "Seven to One". **(B)** Wafer illuminated through photomask "Cell Talk". The images clearly show well-resolved structures, even in the case of the Cell Talk geometries, where the spacing of rectangles was only 1 µm (upper right and lower right of panel (B)). Images recorded with an upright bright-field microscope (Leica DLM 4000, Leica microsystems, Wetzlar, Germany). Scale bars 100 µm.

In the case of 30 µm deep masters, the clean room process was more cumbersome. Next to optimizing the exposure time for the type-3025 SU-8 negative photoresist to 12 seconds, the most crucial step was within the photo development process. As the depth of the cavities (30 µm) was relatively large with respect to the width of the smallest rectangles (10 µm), simple shaking of the wafers in photo developer did not yield sufficient results (Figure 6-2). We here had to introduce an extra step, during which the wafer was sonicated in photo developer for five minutes. Figures 3-2-A and 3-2-B show images of 30 µm deep photoresist masters with patterns "Seven to One" and "Cell Talk", respectively. Clearly, the optimized microfabrication process yielded well-resolved rectangular structures for 30 µm deep photoresist layers.

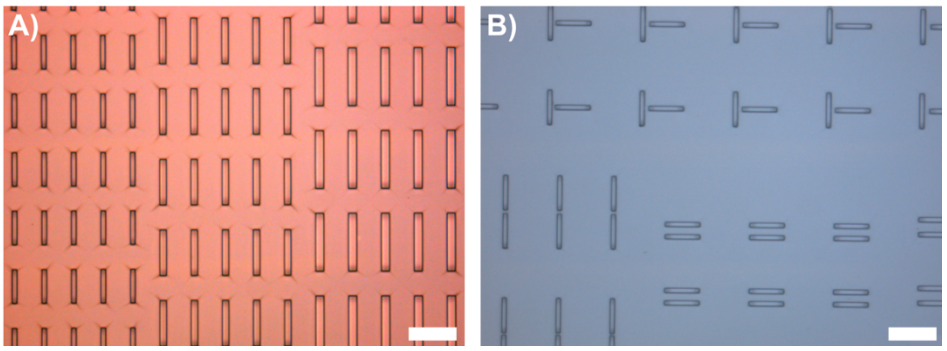

Figure 3-2: 30 µm deep photoresist masters. (A) Wafer illuminated through photomask "Seven to One". **(B)** Wafer illuminated through photomask "Cell Talk". The images show well-resolved rectangular structures. Images recorded with an upright bright-field microscope (Leica DLM 4000, Leica microsystems, Wetzlar, Germany). Scale bars 100 µm.

Figures 3-3-A-D show confocal images of an 8 µm deep and a 30 µm deep PDMS stamp with patterns "Seven to One" at the end of the microfabrication process. For both stamps both, x-y scans and x-z scans, are shown. The x-y scans indicate smooth and well-shaped stamp faces in both cases, and the x-z scans indicate stamp depths of 8 µm and 34 µm, respectively. The intended depth in the case of the 30 µm deep stamps was thus slightly exceeded. However, as the intended 30 µm rather displayed a minimal depth to fabricate robust epoxy membranes (Section 2.3.5), and as the resulting thickness of epoxy membranes yielded good results in the micropatterning process, we decided to work with these 34 µm deep photoresist masters and stamps.

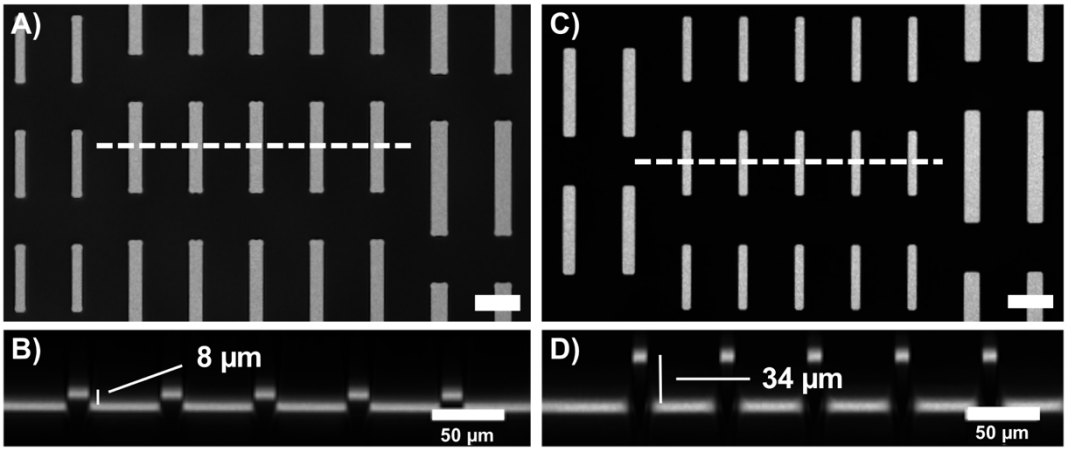

Figure 3-3: Final PDMS stamps at the end of the microfabrication process. (A) x-y scan of an 8 µm deep stamp with pattern "Seven to One". **(B)** x-z scan of the 8 µm deep stamp, as indicated by the dashed line in (A). **(C)** x-y scan of a 30 µm deep stamp with pattern "Seven to One". **(D)** x-z scan of the 30 µm deep stamp, as indicated by the dashed line in (C). The depth of the stamp was 34 µm, and thus a little larger than the intended 30 µm. However, as the intended depth of 30 µm rather displayed a minimal depth in order to fabricate robust enough epoxy membranes (Section 2.3.5), and as this thickness of epoxy membranes yielded good results, we decided to work with these 34 µm deep photoresist masters and stamps. Confocal images were recorded collecting reflected light through a 10X air objective with the pinhole set to 1 airy. Scale bars 50 µm.

3.1.2 Micropatterning of hPSC-derived Cardiomyocytes on Glass Substrates

With the optimized microfabrication protocol, microcontact printing on glass substrates via conventional PDMS stamping (Section 2.3.4) was highly efficient and reproducible. Here, the main difficulty was to determine the proper number of cells to be seeded per area of stamping region/culture dish size. The challenge was to find the right balance between having a good amount of protein islands occupied with cells on the one hand, but not obtaining double or multi occupations on the other hand. Figure 3-4 shows hiPSC-CM on micropatterned glass substrates. Cells were fixed and immunostained for α-actinin 2 using anti-ACTN2 primary antibodies from mouse and Oregon Green 488-conjugated anti-

mouse secondary antibodies. In Figures 3-4-A and 3-4-B, 90,000 cells were seeded per ibidi μ-dish, in Figures 3-4-C and 3-4-D 160,000 cells were seeded. In Figure 3-4-A, roughly 61% of the protein islands were occupied with cells and only few double occupations occurred. In Figure 3-4-C, almost all protein islands were occupied (~ 97%), however, in roughly 38% there were double or multi occupations. As the size of ibidi μ-dishes was 9.6 cm^2 in area, and the final stamps were roughly 1 cm^2, 90,000 cells and 160,000 cells per dish corresponded to cell densities of about 9.4 x 10^3 cells/cm^2 and 16.7 x 10^3 cells/cm^2 (per area of culture dish) or 90 x 10^3 cells/cm^2 and 160 x 10^3 cells/cm^2 (per stamp area), respectively. Cell densities of 9.4 x 10^3 cells/cm^2 per area of culture dish consistently yielded good results and seemed to be the proper cell density to meet the requirements of our experiments.

All in all, Figure 3-4 indicates efficient microcontact printing on glass substrates and good inhibition of cell adhesion to non-printed areas using PLL-g-PEG.

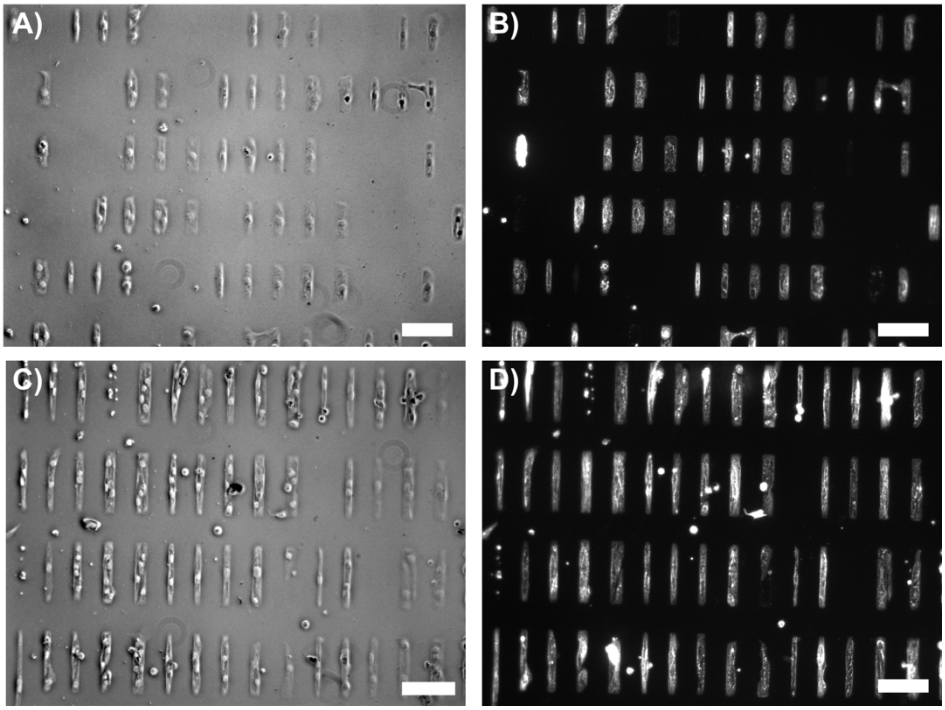

Figure 3-4: hiPSC-CM on Synthemax® micropatterns on glass substrates. Cells were fixed and immunostained for α-actinin 2 (right-hand images). Images were recorded at the center of the stamped area. **(A), (B)** 90,000 cells per ibidi μ-dish. **(C), (D)** 160,000 cells per ibidi μ-dish. Note that the cell patterns shown here were used during the earlier stages of this project, when we had not yet decided on the final shapes of the cardiomyocytes. Scale bars 100 μm.

3.1.3 Micropatterning of hPSC-derived Cardiomyocytes on 15 kPa Elastomer Substrates

On relatively soft 15 kPa elastomer substrates, microcontact printing via conventional PDMS stamping with either 8 μm or 30 μm deep stamps did not yield sufficient results (Figure 6-3). When placing the PDMS stamps onto the substrates, the stamps immersed into the elastomer, which, most likely, resulted in formation of contact and thus protein transfer between the stamps and the substrates not only at the defined stamp faces, but also at the side walls of the stamp faces as well as at the level planes of the stamps. Going to 30 μm deep stamps eliminated protein transfer from the level planes of the stamps, and thus yielded slightly better results, however, protein transfer still occurred at the side walls of the stamp faces.

As described in Section 2.3.5, we thus turned to a method described by Hampe et al. (2014), in which the ECM proteins are deposited through cavities in microstructured lift-off epoxy membranes. Employing this technique, we were able to conveniently micropattern hPSC-CM on 15 kPa elastomer substrates (Figure 3-5). However, at the time when the data for this thesis were recorded, micropatterning on 15 kPa elastomer substrates was not as reproducible as on glass, as in many occasions the epoxy membranes would stick strongly to the elastomer substrates, which resulted in the destruction of parts of the elastomers when trying to lift off the epoxy membranes. We thus had to limit our experiments to intact regions of the substrates[7].

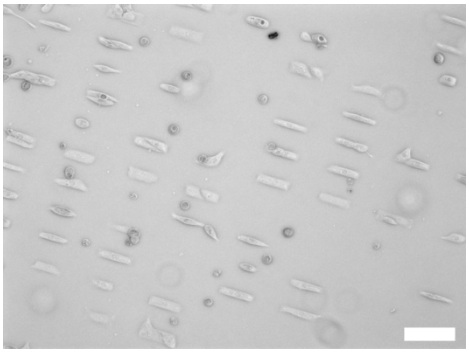

Figure 3-5: hiPSC-CM on Synthemax® micropatterns on 15 kPa elastomer substrates. Cells were micropatterned using microstructured lift-off epoxy membranes, as first described by Hampe et al. (2014). The cell patterns shown here were used during the earlier stages of this project, when we had not yet decided on the final shapes of the cardiomyocytes. Scale bar 70 μm.

[7] As the project has continued behind the scope of this thesis, we have resolved this problem of strong adhesion between the epoxy membranes and the elastomers by i) storing the uncured epoxy resin at 4 °C, and ii) increasing the UV illumination time during the curing of the epoxy resin.

3.2 CRISPR/Cas9-mediated Endogenous Tagging

As described in Section 2.2, we aimed to establish a genome-edited HES-2-derived cell line, in which Citrine YFP was fused to the C-terminus of α-actinin 2. Editing the genome at the stem cell level, and exploiting our group's expertise in the differentiation of stem cells into CM, yielded an, in principle, unlimited number of CM expressing the fusion protein ACTN2-Citrine, thus an endogenous label of the sarcomeric network.

3.2.1 CRISPR/Cas9 Efficiency

24 hours after co-transfection of wild-type HES-2 (HES-2-WT) with the gRNA/Cas9 and donor DNA plasmids, cells were analyzed and single-cell sorted into 96-well plates via FACS. In our CRISPR plasmids, a GFP marker was fused to the Cas9, such that cells expressing Cas9 co-expressed a GFP and could be selected by their fluorescence signal. Figure 3-6 shows typical scatter plots and gating from FACS, here exemplified for the case of gRNA-1 + donor DNA "Linker". Between all gRNA/donor pairs used in this project, transfection efficiency varied from 1.8% to 2.9% and was thus fairly low, however, at least 96 GFP-positive cells could be harvested for each gRNA/donor pair, respectively, to grow a sufficient number of single-cell clones.

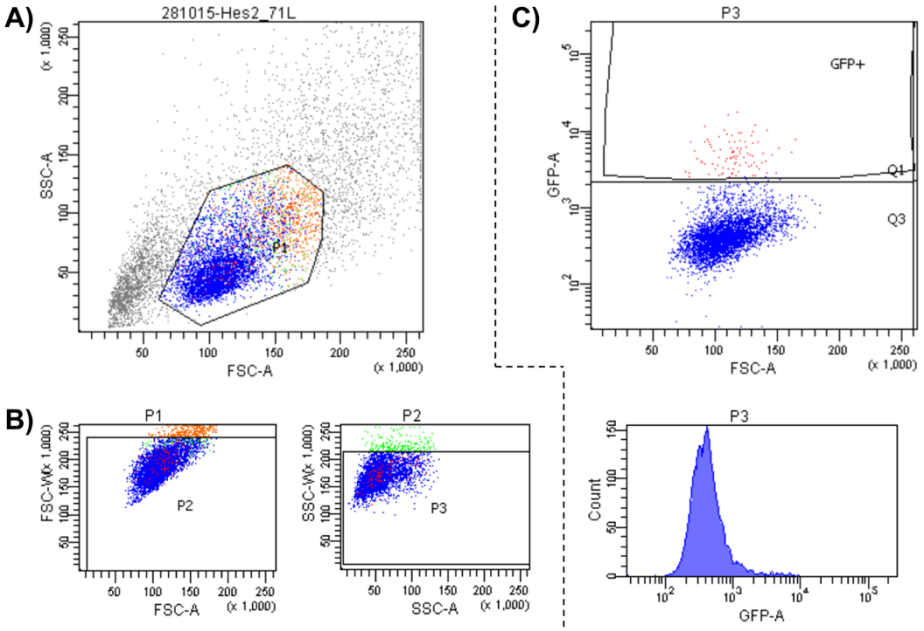

Figure 3-6: Single-cell sorting of HES-2 expressing GFP-labeled Cas9. After co-transfection of wild-type HES-2 with gRNA/Cas9 and donor plasmids, GFP-positive cells were single-cell FACS sorted into 96-well plates. FACS work flow and settings are here exemplified for the case of gRNA-1 + donor DNA "Linker". Side scatter vs. forward scatter plots were used to identify the cell cohort of interest. Further gating in forward scatter width vs. forward scatter area plots as well as side scatter width vs. side scatter area plots was conducted to discriminate doublets. In the final P3 gate, Cas9-positive cells were selected via GFP intensity. FSC-A/W: forward scatter area/width; SSC-A/W: side scatter area/width; GFP-A: GFP area.

One week after seeding into 96-well plates, the clones were checked by eye using a binocular to test for obvious polyclonality. Growth of clones could be observed in 16 to 27 wells of each 96-well plate, respectively, corresponding to a total of 87 clones. In 14 to 20 of these clones no obvious polyclonality was observed. One week later, i.e. two weeks after seeding, only 81 of the 87 clones had continued to grow sufficiently, and these 81 clones were PCR genotyped to examine successful integration of Citrine into the ACTN2 gene locus (Section 2.2.3, primers listed in Table 6-2). Figure 3-7 shows the resulting electrophoresis gels for the three clones which eventually were taken into differentiation and one representative homozygous clone. Table 3-1 summarizes the results for all clones. PCR genotyping confirmed successful integration of Citrine (homo- or heterozygous) in 19 out of 81 analyzed clones. Heterozygous integration was observed in 15 clones, and homozygous integration in four clones. For eight clones, DNA gel electrophoresis yielded unexpected band patterns with fragment lengths other than ~ 1000 or ~ 1700 bp (data not shown). Notably, the growth of all four clones with homozygous integration was significantly slower than that of all others, and none of these clones were grown to amounts sufficient for differentiation into CM.

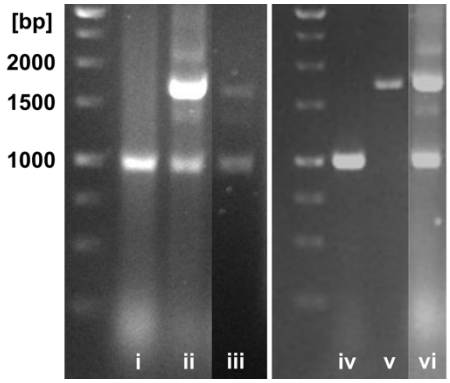

Figure 3-7: PCR genotyping of representative knock-in clones. PCR and gel electrophoresis were performed two weeks after FACS. **i)** Negative control. **ii)** Clone gRNA-1-SerGG-A8: presumably heterozygous integration of Citrine but polyclonal (see below). **iii)** Clone gRNA-2-4Mut-G6: heterozygous. **iv)** Negative control. **v)** Clone gRNA-1-Linker-A8: homozygous. **vi)** Clone gRNA-1-Linker-G6: heterozygous.

Table 3-1: Genotyping of CRISPR clones.

gRNA/donor	# clones genotyped	hom.	het.	unspecific	neg.
gRNA-1/SerGG	16	0 (0.0%)	1 (6.3%)	2 (12.5%)	13 (81.3%)
gRNA-1/Linker	23	3 (13.0%)	4 (17.4%)	0 (0.0%)	16 (69.6%)
gRNA-2/4Mut	22	0 (0.0%)	5 (22.3%)	2 (9.1%)	15 (68.2%)
gRNA-2/Linker	20	1 (5.0%)	5 (25.0%)	4 (20.0%)	10 (50.0%)

Two weeks after FACS sorting, all sufficiently grown clones were PCR genotyped to monitor integration of Citrine into the ACTN2 gene locus (Figure 3-7). PCR genotyping confirmed homozygous or heterozygous integration in 19 of the 81 analyzed clones. In the case of eight clones, as listed in column "unspecific", we observed an unexpected band pattern, with DNA fragment lengths other than ~ 1000 or ~ 1700 bp.

3.2.2 Imaging of Bulk HES-2-ACTN2-Citr-derived Cardiomyocytes

Cells of three clones were eventually taken into cell culture for differentiation into cardiomyocytes. The first cell line, termed *HES-2-Citr-Linker*, was produced using gRNA-1 and donor DNA Linker. The second cell line, termed *HES-2-Citr-4Mut*, was obtained using gRNA-2 and donor DNA 4Mut. The third cell line, termed *HES-2-Citr-SerGG*, was engineered using gRNA-1 and donor DNA SerGG. This third cell line, however, turned out to be polyclonal (Section 3.2.4), and was not further used for experiments.

Figure 3-8 shows bright-field and epifluorescence images of CM derived from each, HES-2-Citr-4Mut (HES-2-Citr-4Mut-CM) and HES-2-Citr-Linker (HES-2-Citr-Linker-CM). The

images were recorded roughly four weeks after start of differentiation, and in the T-25 culture flasks that were used during the differentiation process. As the objective lenses of the microscope were not optimized for the thick plastic bottom of the T-25 flasks, the image quality was relatively poor. However, the epifluorescence images clearly indicate bright fluorescence signal with the exposure time set to only 250 ms. In fact, the signal was bright enough to record time series at up to 10 fps even under these poor imaging conditions (video data not shown). Notably, clear formation of organized myofibrils could be observed in regions with lower cell density and clear ACTN2 striation patterns became visible.

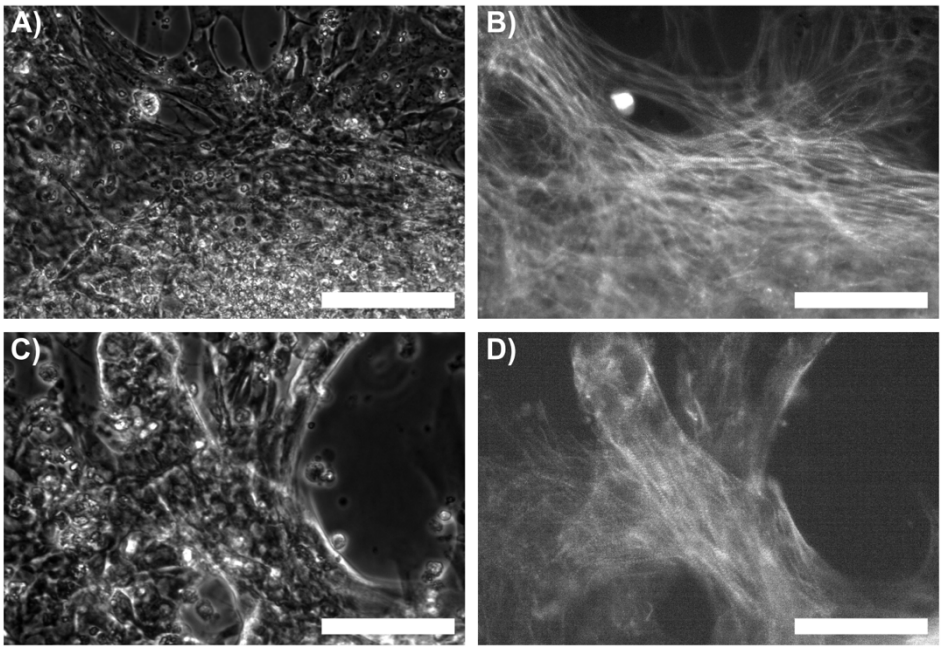

Figure 3-8: Bulk HES-2-ACTN2-Citr-CM. Bright field and respective fluorescence images for line HES-2-Citr-4Mut **(A+B)** and for line HES-2-Citr-Linker **(C+D)**. Images were recorded in T-25 culture flasks roughly four weeks after the start of differentiation and prior to the first digest. Exposure time was 250 ms in the case of the fluorescence images. Scale bars 100 μm.

To gain more detailed insight into sarcomere dynamics, we turned to high-speed confocal video imaging by employing our 8 kHz resonant scanning confocal microscope (Section 2.4.1). Figure 3-9-A shows a single frame from a typical video of a 2-D monolayer of HES-2-ACTN2-Citr-CM. CM, here derived from HES-2-Citr-4Mut, were seeded to confluent monolayers on Synthemax®-coated glass. Figures 3-9-B and 3-9-C display detail views of the marked area in Figure 3-9-A at two different time points. In Figure 3-9-B, the network of CM/sarcomeres was at rest, whereas n Figure 3-9-C, the (synchronized) sarcomeres were at peak contraction. The red lines marking two Z-lines clearly indicate movement of Z-lines during contraction. Displacement of the labeled two Z-lines in between frames could be estimated to roughly 0.5 μm and 0.9 μm, respectively. To illustrate the periodic movement of Z-lines over time, Figure 3-9-D shows a kymograph of several sarcomeres

labeled by the pink line in Figure 3-9-A (here ~ 7 seconds displayed; small inset shows kymograph for 25 seconds). Clearly, the sarcomere contractions occurred highly periodically, and the movement of Z-lines seemed highly homogenous.

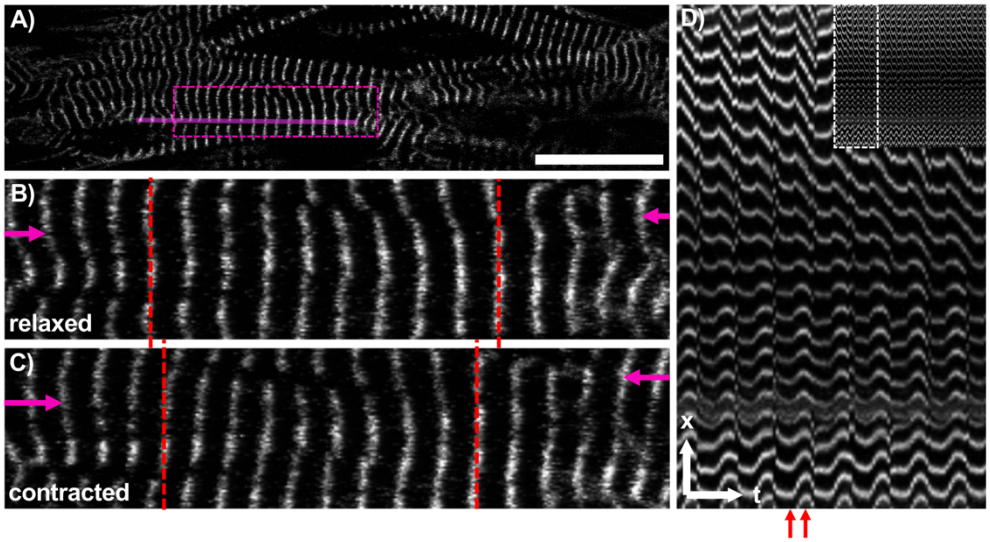

Figure 3-9: Confocal video imaging of bulk HES-2-ACTN2-Citr-CM. CM, here derived from HES-2-Citr-4Mut, were seeded to confluent monolayers on Synthemax®-coated glass. Time series images were recorded five days after seeding and at 14.5 fps. **(A)** Overview image of CM at relaxed state. **(B), (C)** Detailed views of area marked by pink rectangle in (A). **(B)** Relaxed state. **(C)** Sarcomeres at peak contraction. Pink arrows and red lines mark several same Z-lines for each time point, respectively, to illustrate movement of Z-lines during contraction. **(D)** Kymograph illustrating periodic displacement of Z-lines over time. Plotted Z-lines are indicated by pink line in (A). Intensity was averaged over 10 scan lines in the y-direction. For illustration purposes, only 7 of 25 seconds of recorded beating are displayed (compare small inset). Red arrows mark time points as shown in (B) and (C). Scale bar 20 µm.

3.2.3 Micropatterning of HES-2-ACTN2-Citr-derived Cardiomyocytes

Figure 3-10 shows confocal images of live HES-2-ACTN2-Citr-CM on micropatterned glass (Figure 3-10-A) and 15 kPa elastomer substrates (Figure 3-10-B). CM in panel A were derived from HES-2-Citr-Linker, whereas CM in panel B were derived from HES-2-Citr-4Mut. A high number of fluorescent, analyzable cells was available for parallel recordings.

It became evident that the cells in the center five rows, which were of sizes 70 µm x 10 µm, expressed more organized myofibrils as compared to the larger cells in the top and bottom rows (122 µm x 17 µm and 95 µm x 13.5 µm, both also aspect ratio 7:1). As this could be consistently observed throughout all our experiments, we limited our analysis to cells of size 70 µm x 10 µm = 700 μm^2, which resembles the spread area typically reported for hPSC-CM in 2D culture and the aspect ratio found in adult cardiomyocytes (Section 1.1). The overall level of sarcomere organization on the 15 kPa substrate in Figure 3-10-B

seemed lower, as compared to the glass substrate in Figure 3-10-A, but notably, this impression only held true for regions with poor micropatterning on the elastomer. At the time when the data for this thesis were recorded, micropatterning on 15 kPa substrates was not as reproducible as on glass, since in many occasions, lifting the epoxy membranes off the elastomer substrates resulted in the destruction of parts of the elastomer, as described in Section 3.1.3. We thus had to limit our data recording to the intact regions of the substrates, however, even in partly disrupted elastomer regions, like the one shown in Figure 3-10-B, there were some well-assembled and analyzable cells, as indicated by the red arrow.

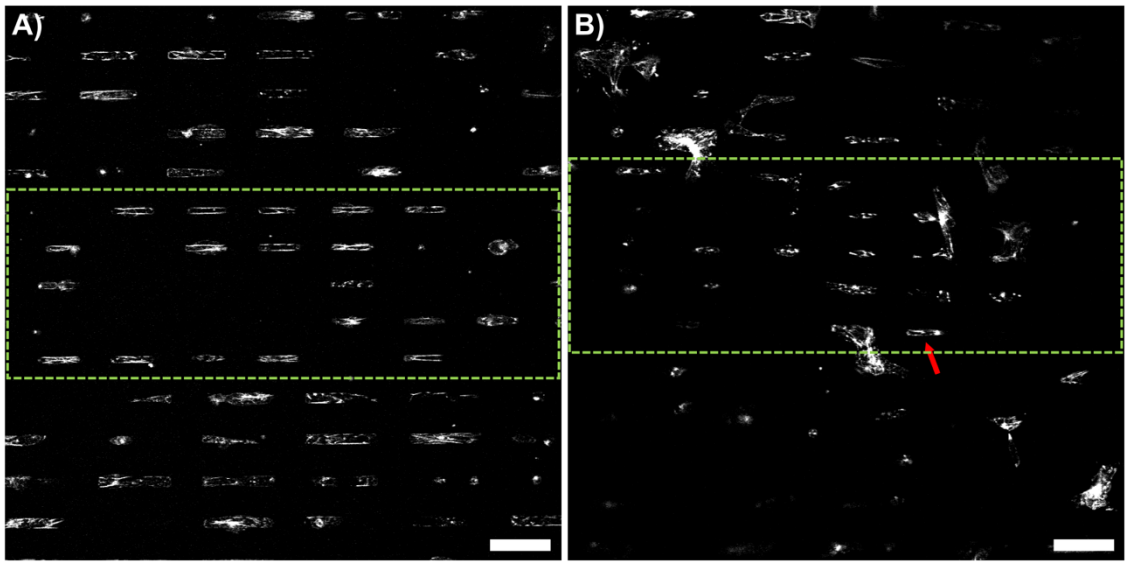

Figure 3-10: Micropatterning of live HES-2-ACTN2-Citr-CM. (A) CM derived from HES-2-Citr-Linker on a glass substrate. **(B)** CM derived from HES-2-Citr-4Mut on a 15 kPa elastomer substrate. Micropatterning was pursued using PDMS stamps with pattern "Seven to One" (i.e. three different cell sizes at aspect ratio 7:1). Center five rows (marked by green box) show cells with typical sizes 70 µm x 10 µm, which were used for experiments. Red arrow indicated intact CM despite locally disrupted elastomer (see text for further remarks). Scanning images were recorded using a 10X air objective. Scale bars 100 µm.

Figure 3-11 shows close-up confocal images of single micropatterned HES-2-ACTN2-Citr-CM on glass and 15 kPa substrates for both, HES-2-Citr-Linker (Figure 3-11-Aa and-Ab) and HES-2-Citr-4Mut (Figure 3-11-Ba and -Bb). Confocal images were recorded using a 63X immersion oil objective, and display single frames from video imaging at 18.2 fps. Clearly, well-organized sarcomeric networks could be observed for both cell lines and on both types of substrates. Together with Figure 3-10, these images demonstrate successful micropatterning of HES-2-ACTN2-Citr-CM.

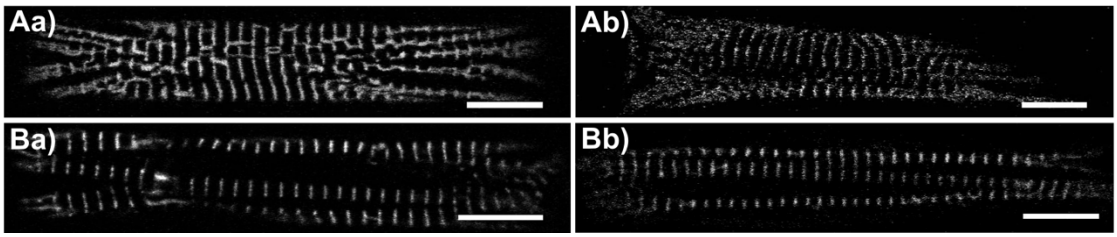

Figure 3-11: Confocal imaging of micropatterned live HES-2-ACTN2-Citr-CM. Images display single frames from video imaging at 18.2 fps (Section 3.3). **(Top)** CM derived from HES-2-Citr-Linker on a glass substrate **(Aa)** and on a 15 kPa elastomer substrate **(Ab)**. **(Bottom)** CM derived from HES-2-Citr-4Mut on glass **(Ba)** and 15 kPa **(Bb)**. Scale bars 10 μm.

3.2.4 Immunolabeling of HES-2-ACTN2-Citr-derived Cardiomyocytes

To test for the morphologically correct integration of Citrine and thus for the correct morphologic labeling of Z-lines in HES-2-ACTN2-Citr-CM, we turned to immunolabeling. HES-2-ACTN2-Citr-CM were seeded on micropatterned substrates, fixed and co-stained for actin (phalloidin-Atto550) and α-actinin 2 (anti-ACTN2 from mouse, anti-mouse-Alexa633 from goat). Figure 3-12 shows close-up confocal images of a single cardiomyocyte derived from HES-2-Citr-Linker on a micropatterned glass substrate. Panel B shows the endogenous fluorescence signal from Citrine, panel C represents actin, and channel D represents α-actinin 2. Panel A shows the merged image of all three fluorescence channels. Panels (B) and (D) indicate clear co-localization of Citrine and ACTN2. In panel (C), h-zones become clearly visible, and thus ACTN2 and Citrine, which co-localize with the bright lines in the actin channel, are located at the Z-lines, as intended.

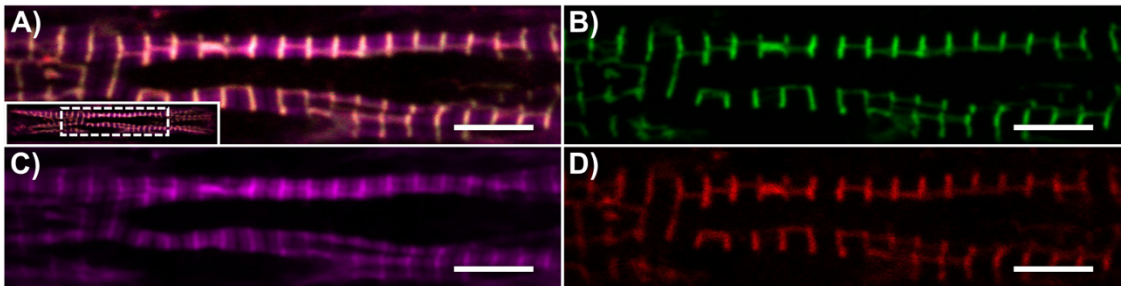

Figure 3-12: Correct morphological integration of Citrine. Micropatterned HES-2-ACTN-Citr-CM were fixed and immunostained for actin **(C)** and α-actinin 2 **(D)**. **(B)** shows fluorescence signal of Citrine YFP. **(A)** shows merged images of all three fluorescence channels. Panels (B) and (D) indicate clear co-localization of Citrine and ACTN2. In panel (C), h-zones become clearly visible, and thus ACTN2 and Citrine are located at the Z-lines, as expected. CM shown here were derived from HES-2-Citr-Linker. The three different fluorescence channels were scanned sequentially to minimize cross talk. Scale bars 5 μm.

Immunolabeling further gave insight into the presence of the several cell populations within each differentiation, and also potentially into polyclonality within the lines. Figure 3-13 shows confocal overview images of CM derived from both, HES-2-Citr-Linker (Figures 3-13-Aa-c) and from HES-2-Citr-SerGG (Figures 3-13-Ba-c) on micropatterned glass substrates. Channel a shows the endogenous fluorescence signal from Citrine, channel b represents actin, and channels c represents immunostained α-actinin 2. As actin was expressed in both myocytes and non-myocytes, whereas α-actinin 2 as well as the fusion protein ACTN2-Citrine were only expressed in myocytes, the ratio of Alexa633-positive cells over Atto550-positive cells represented the fraction of myocytes present in the field of view. The ratio of Citrine-positive cells over Alexa633-positive cells represented the fraction of endogenously labeled myocytes. As one would expect all myocytes to express the ACTN2-bound Citrine in a monoclonal cell line, and assuming complete penetrance, a fraction of Citrine-positive myocytes < 1 indicated polyclonality in the cell cohort[8]. In the case of HES-2-Citr-Linker, a total of n = 183 cells was analyzed in three different, randomly chosen field of views, and in the case of HES-2-Citr-SerGG a total of n = 123 cells was analyzed. For HES-2-Citr-Linker, the purity of cardiomyocytes was 90.7% (166/183) in this particular differentiation. 100% of those 166 myocytes were positive for Citrine, thus no signs of polyclonality could be observed via immunolabeling. For HES-2-Citr-SerGG, the purity of myocytes was only 72.4% (89/123) and only 87.6% (78) of those 89 myocytes were positive for Citrine. Our experiments thus suggested polyclonality or incomplete penetrance in the cell line HES-2-Citr-SerGG and therefore this cell line was not further used for experiments. Notably, we did not run similar experiments for the cell line HES-2-Citr-

[8] This holds true even in a heterozygous knock-in, as one Z-disc comprises several hundreds of actinins, and thus the likelihood for a knock-in CM to express all actinins from the untargeted allele is negligible.

4Mut. However, in all our *contractility experiments* (Sections 3.3 and 3.4) we did not observe a single HES-2-Citr-4Mut-CM that did not exhibit Citrine fluorescence, and also flow cytometry analysis (data not presented) indicated 100%-positivity for Citrine for both, CM derived from HES-2-Citr-4Mut as well as HES-2-Citr-Linker.

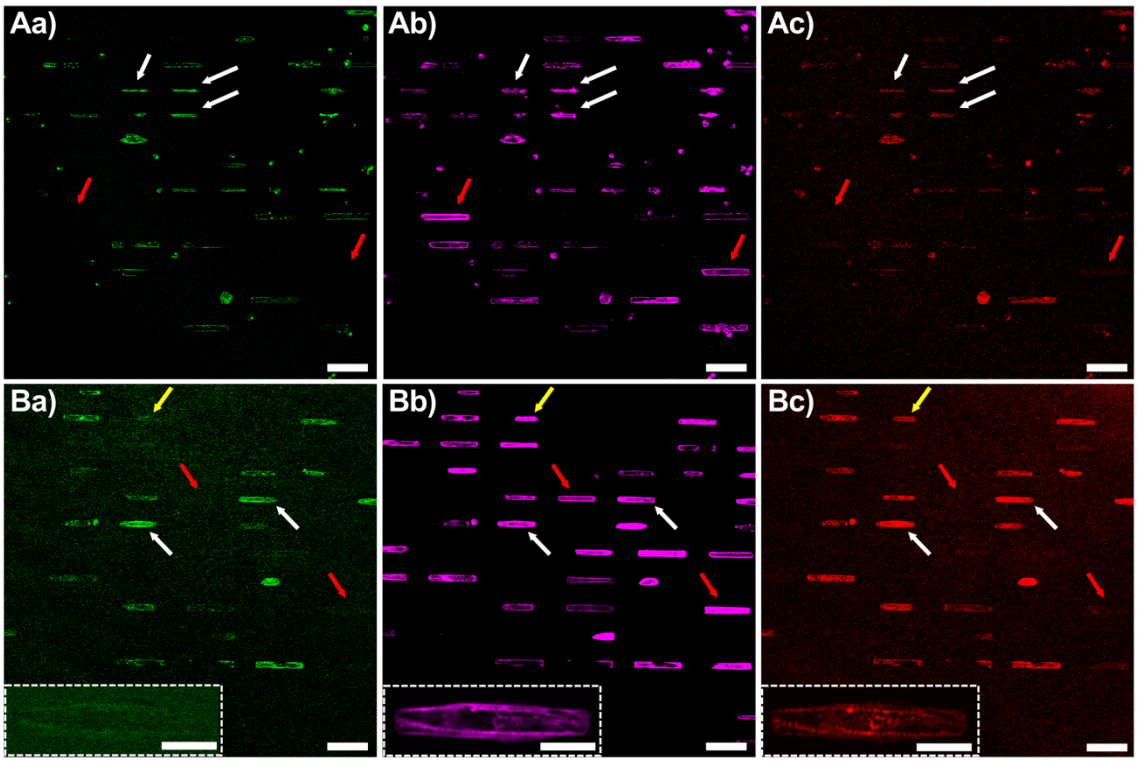

Figure 3-13: Anti-ACTN2 and anti-actin co-staining of HES-2-ACTN2-Citr-CM. **(A)** CM derived from HES-2-Citr-Linker on a micropatterned glass substrate. **(B)** CM derived from HES-2-Citr-SerGG on a micropatterned glass substrate. Cells were fixed and immunostained for actin **(channel b)** an α-actinin 2 **(channel c)**. **Channel a** shows the endogenous fluorescence signal of Citrine YFP. White arrows mark respresentative cardiomyocytes (positive for actin and α-actinin 2). Red arrows mark non-myocytes (positive for only actin but not for α-actinin 2). Yellow arrows in (B) mark myocyte positive for both actin and α-actinin 2, but without Citrine fluorescence, as also displayed in the insets in (B). The presence of CM without endogenous Citrine fluorescence indicated polyclonality or incomplete penetrance in the cell line HES-2-Citr-SerGG. Scanning images were recorded using a 10X air objective. Scale bars 100 μm.

3.2.5 Western Blotting of HES-2-ACTN2-Citr-derived Cardiomyocytes

For further analysis of protein expression we performed immunoblots for α-actinin 2 and Citrine. Immunoblots were performed on cell lysates of CM derived from the two cell lines HES-2-Citr-Linker and HES-2-Citr-4Mut, and on cell lysates of CM derived from HES-2-WT for control (Figure 3-14).

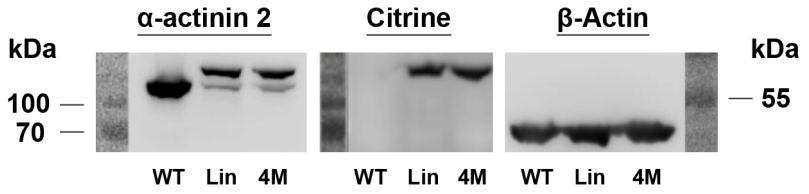

Figure 3-14: ACTN2 and Citrine immunoblots of transgenic cell lines. SDS-PAGE immunoblots for α-actinin 2 and Citrine were performed on lysates of CM derived from transgenic cell lines HES-2-Citr-Linker (Lin) and HES-2-Citr-4Mut (4M) as well as on lysates of CM derived from HES-2-WT (WT) for control. Immunoblots for β-actin as loading control.

ACTN2 and Citrine immunoblots confirmed the expression of wild-type sized ACTN2 in all three cell lines, whereas expression of Citrine was only observed in the two mutant cell lines HES-2-Citr-Linker and HES-2-Citr-4Mut. Next to the wild-type sized ACTN2 bands, ACTN2 immunoblots further showed a second and slightly longer protein band in the case of the two knock-in cell lines. This band correlated with a protein roughly 25 kDa larger in size, as compared to wild-type ACTN2, and thus corroborated heterozygous expression of the fusion protein ACTN2-Citrine, as the size of Citrine YFP is known to be roughly 27 kDa (Prendergast and Mann 1978).

Strikingly, the ACTN2 immunoblots of both transgenic cell lines suggested higher expression levels for the targeted as compared to the untargeted allele. Densitometry analysis (targeted/untargeted) yielded expression levels of roughly 80% for the knock-in allele in CM derived from both lines.

3.2.6 Genomic Sequencing of HES-2-ACTN2-Citr

To further corroborate the correct integration of Citrine into the ACTN2 gene locus, we performed DNA sequencing on DNA fragments flanking the predicted Cas9 cleaving site by roughly ± 500 bp. DNA fragments of the cell lines HES-2-Citr-Linker, HES-2-Citr-4Mut, and HES-2-WT (for control) were amplified via PCR using the same primers as used for genotyping (Table 6-2). Both, the knock-in alleles and the untargeted alleles of the knock-in cell lines, as well as the WT alleles for control, were sent in for DNA sequencing (Section 6.4). The detailed sequencing results are listed in Section 6.2.3.

DNA sequencing confirmed the correct and heterozygous integration of Citrine into the ACTN2 gene locus of both, HES-2-Citr-Linker and HES-2-Citr-4Mut (Figure 3-15). Citrine was inserted in front of the endogenous STOP codon of the last ACTN2 exon and the flanking nucleotides within the 5' and 3' homology arms were unaltered (full sequencing results in Section 6.2.3).

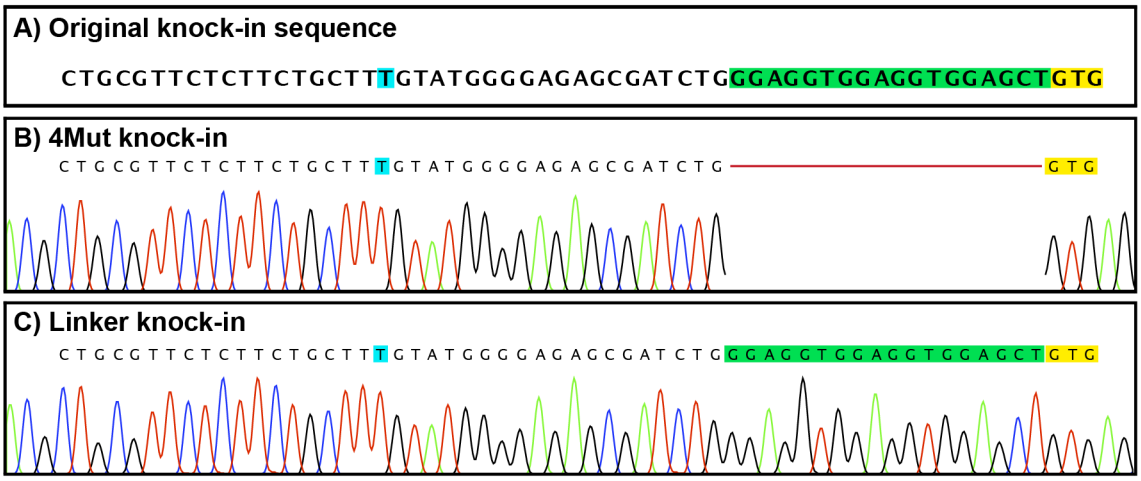

Figure 3-15: DNA sequencing of transgenic knock-in alleles. (A) Original donor DNA sequence, here including linker sequence (marked in green). **(B)** HES-2-Citr-4Mut knock-in allele. **(C)** HES-2-Citr-Linker knock-in allele. Chromatograms are shown only for the nucleotides flanking the predicted Cas9 cleavage site T. Full sequencing results are listed in Section 6.2.3. GTG marks the start of the Citrine coding sequence.

In the case of the untargeted alleles, DNA sequencing revealed a single nucleotide base pair insertion directly at the Cas9 cleaving site (Figure 3-16). This indel lead to a shift of reading frame with an alternative STOP, and resulted in a mutated wild-type allele with an extra 44 amino acids as compared to the endogenous HES-2 wild-type sequence (Section 6.2.3).

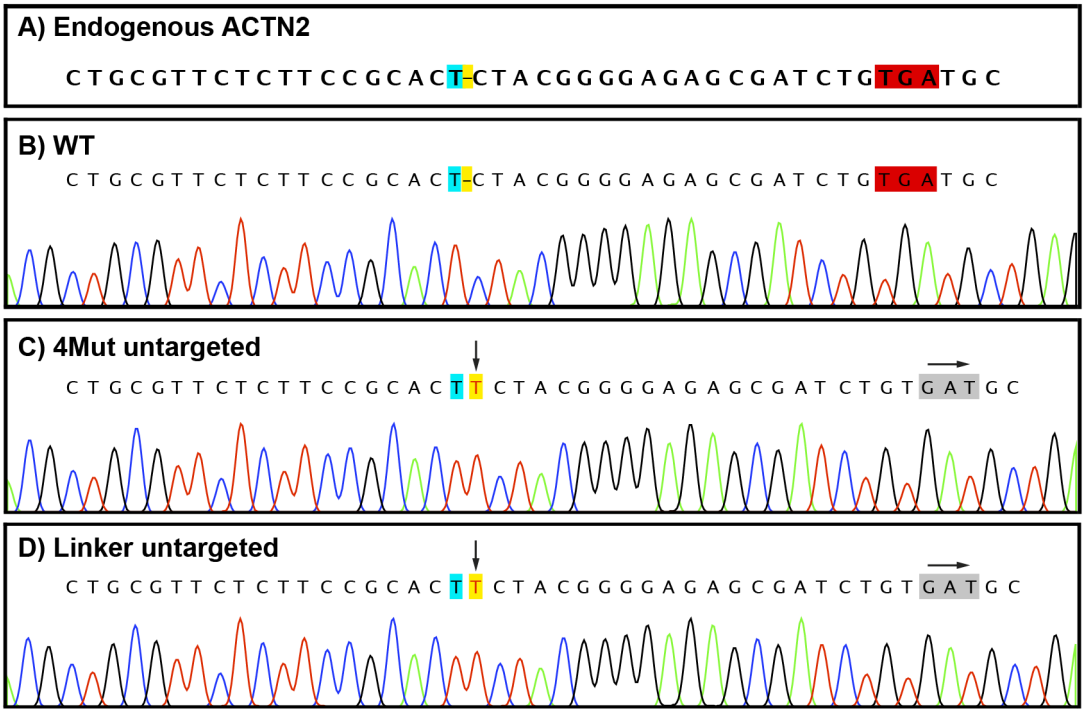

Figure 3-16: DNA sequencing of the untargeted alleles. (A) Endogenous alpha-actinin 2. **(B)** HES-2-WT for control. **(C)** HES-2-Citr-4Mut untargeted allele. **(D)** HES-2-Citr-Linker untargeted allele. A single base pair insertion T directly at the Cas9 cleaving site T was observed in the untargeted alleles of both transgenic cell lines. This small indel resulted in a shift of reading frame, which lead to a misread of the endogenous STOP codon TGA, and in turn to a mutated wild-type allele with an extra 44 amino acids, as compared to the endogenous HES-2-WT sequence. Chromatograms are shown only for the nucleotides flanking the predicted Cas9 cleavage site. Full sequencing results are listed in Section 6.2.3.

3.2.7 Protein Structure Analysis

To evaluate potential folding perturbation induced by either the extra 44 AS in the mutated untargeted allele (Section 3.2.6), or induced by the Citrine, we used the web-based Phyre2 platform to predict the 3D protein structures of the three ACTN2 variants of interest, and compared them to the wild-type ACTN2 conformation (Kelley et al. 2015).

Comparison of the mutated ACTN2 from the untargeted allele (Figure 3-17-B) and the wild-type form (Figure 3-17-A) indicated that the extra 44 amino acids, resulting from the frame shift mutation (Section 3.2.6), added an extra alpha-helix to the C-terminal of the protein (green arrow). Seemingly this added alpha-helix left the other domains of the protein, particularly the N-terminal actin binding domain (blue arrow) and C-terminal EF hands, morphologically unaltered.

The fusion of beta-barrel-shaped Citrine to the C-terminus of ACTN2 was predicted to interfere with the folding of ACTN2, where it seemed to induce a bending of the alpha-

helices located at the far C-terminus (Figure 3-17-C and Figure 3-17-D). This bending did not appear to depend on whether the two proteins were fused via a linker peptide or directly (white and red arrows), however, direct fusion without a linker peptide seemed to induce further conformational changes in the alpha-helices at the very far C-terminal end.

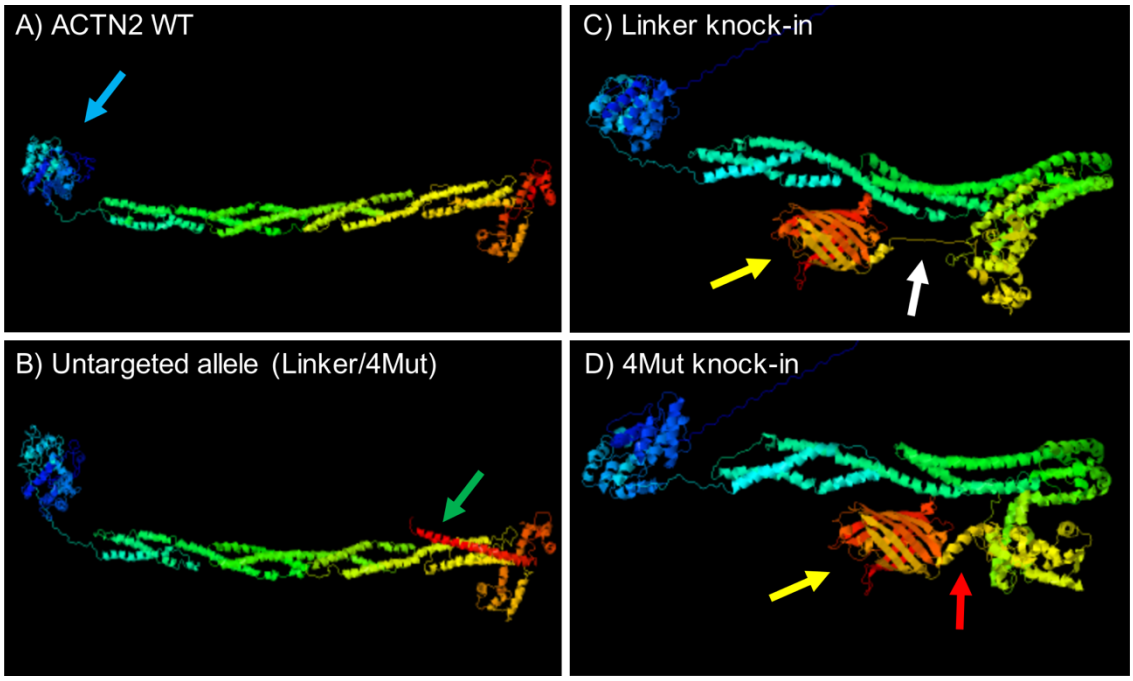

Figure 3-17: 3D protein models. (A) Wild-type α-actinin 2. Blue arrow marks actin binding domain. **(B)** Mutated ACTN2 expressed via untargeted allele, as present in both, HES-2-Citr-Linker and HES-2-Citr-4Mut. Green arrow marks alpha-helix, which comprises an extra 44 AS induced by the frame-shift mutation in the untargeted allele. **(C)** ACTN2-Citrine fusion protein as predicted for HES-2-Citr-Linker. Yellow arrow marks Citrine beta barrel. White arrow marks linker peptide. **(D)** ACTN2-Citrine fusion protein as predicted for HES-2-Citr-4Mut, where no linker peptide was inserted. Red arrow marks ACTN2 alpha-helix which was directly fused to Citrine. Protein models were predicted using the web-based platform Phyre2 (Kelley et al. 2015).

3.3 Contractility of HES-2-ACTN2-Citr-derived Cardiomyocytes

To characterize the sarcomere function in HES-2-ACTN2-Citr-derived CM, we studied the movement of Z-lines in so-called *contractility experiments*. In these contractility experiments, we confocally imaged the spontaneous sarcomeric beating of single micropatterned HES-2-ACTN2-Citr-CM for 20 seconds at 18.2 fps. We then employed a custom-developed video analysis algorithm to extract basics characteristic parameters of cardiomyocyte beating, as well as a measure for the coherence of sarcomere contractions between several sarcomeres within single cardiac myofibrils. All contractility experiments were performed at 37 °C and 5% CO_2.

3.3.1 The Video Analysis Algorithm

To analyze ~ 150 confocal video images, each consisting of roughly 350-400 frames, we custom-developed an analysis algorithm, which did not only yield basic characteristic parameters of cardiomyocyte beating, but also allowed for the evaluation of the coherence of sarcomere contractions within single cardiac myofibrils. Scripting and video analysis was performed in Python® (Section 2.5). In the following, we demonstrate crucial steps of the video analysis algorithm by employing it to one example cell:

Figure 3-18-A shows a single frame from a typical contractility experiment. Figure 3-18-B shows the corresponding kymograph, illustrating periodic displacement of Z-lines. The cardiomyocyte, here derived from HES-2-Citr-4Mut and seeded on a 15 kPa elastomer substrate, was imaged for 20 seconds at 18.2 fps. The time series of 364 images was transferred to ImageJ-based Fiji (Schindelin et al. 2012), and a myofibril, generally comprising at least eight and up to 20 sarcomeres, was selected, as illustrated by the pink line in Figure 3-18-A. The intensity profile in this region of interest (ROI) was measured over time, averaged in the y-direction, and saved into a data spreadsheet. The data spreadsheet was then transferred to Python®, and in each frame, the intensity profile was analyzed for intensity peaks to detect the current positions of Z-lines. A custom-written algorithm was then employed to assign the intensity peaks to Z-line trajectories (Figure 3-18-C), and for each sarcomere, the change in sarcomere length over time, ΔSL, was derived (Figure 3-18-D).

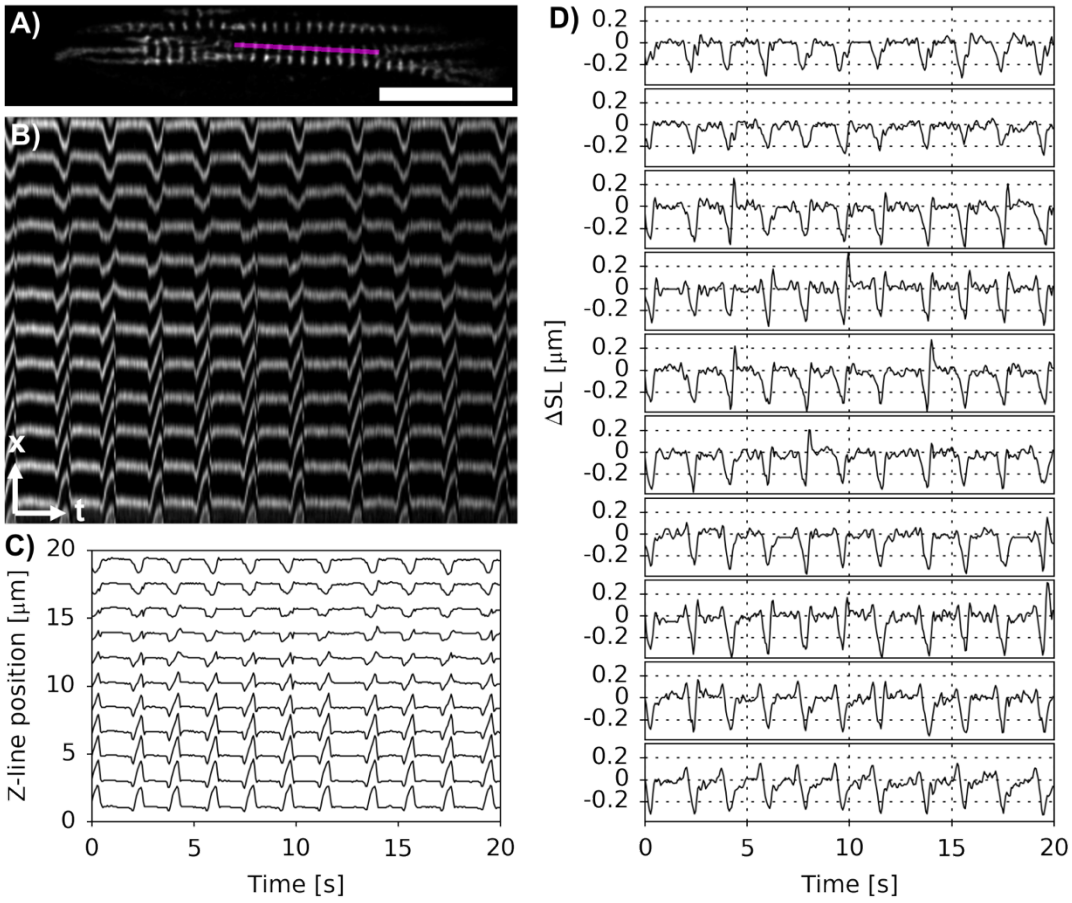

Figure 3-18: Z-line trajectories and sarcomere contraction amplitudes. (A) Cardiomyocyte derived from HES-2-Citr-4Mut on a 15 kPa elastomer substrate. A myofibril, here comprising 11 sarcomeres (corresponds to 12 Z-lines), was selected, as illustrated by the pink line. The intensity profile in the selected ROI was analyzed using our custom-developed algorithm to determine single Z-line trajectories. **(B)** Kymograph illustrating periodic displacement of the 12 Z-lines in the selected ROI in (A). **(C)** Z-line trajectories, as determined via our custom-developed Python® script, mimic the above kymograph. **(D)** Change in sarcomere lengths over time, ΔSL, as derived from Z-line trajectories. For concision purposes, ΔSL is only displayed for 10 sarcomeres in (D), and only the corresponding 11 Z-lines are displayed in (C). Scale bar 20 µm.

Next, the starting time points of each contraction were manually chosen by pointing to the first visible Z-line displacement in the (magnified) Z-line trajectories. These starting points were corrected via cross-correlation between all contractions of each sarcomere, and the corrected starting points for contraction cycle i were defined as *start of contraction*, $t_{start,i}$, as indicated by the red lines in Figure 3-19-B. The beating period, T, was calculated as $T_i = |t_{start,i+1} - t_{start,i}|$, and the beating frequency, f, was calculated as $f = 1/\text{mean}(T_i)$. The sarcomere length at rest, SL_0, was deduced from the average length of each sarcomere at equilibrium, and the minimal sarcomere length, Min. SL, was defined as the minimal sarcomere length observed during the full recording time (Figure 3-19-A). The (peak) contraction

amplitude, ΔSL_{max}, was defined as the change in sarcomere length at maximum contraction, i.e. between t_{start} and t_{max}, where t_{max} is the time point of maximum contraction (black dashed lines in Figure 3-19-B), thus $\Delta SL_{max,i} = |\Delta SL(t_{max,i}) - \Delta SL(t_{start,i})|$ (Figure 3-19-B). The maximal contraction amplitude, Max. ΔSL_{max}, was defined as the greatest ΔSL_{max} observed within each cell during recording time (Figure 3-19-B). The temporal duration until maximum contraction, T_{max}, herein and after referred to as time-to-peak contraction, was calculated as $T_{max,i} = |t_{max,i} - t_{start,i}|$ (Figure 3-19-B). We chose to measure the above-described parameters Min. SL and Max. ΔSL_{max} next to determining the single contraction amplitudes $\Delta SL_{max,i}$ and their average ΔSL_{max}, as we observed that the temporal resolution at 18.2 fps did not suffice to detect subtle changes in ΔSL_{max} in all of our experiments, especially in those where we studied the effects of drug treatment (Section 3.4). Particularly the *minimal* value Min. SL is less prone to get affected by inaccuracies in determining/recording of $t_{max,i}$ and thus $\Delta SL(t_{max,i})$ due to limited temporal resolution. To this end, note that $\Delta SL_{max,i}$ depends on the accuracy of both, $\Delta SL(t_{start,i})$ and $\Delta SL(t_{max,i})$, and therefore on the precision and accuracy in determining $t_{start,i}$ and $t_{max,i}$ (Figure 3-19-B). The accuracy of the *minimal* value Min. SL, however, will grow with the likelihood of capturing the timepoint of the fully contracted sarcomere and thus with the number of observed beatings (Figure 3-19-A).

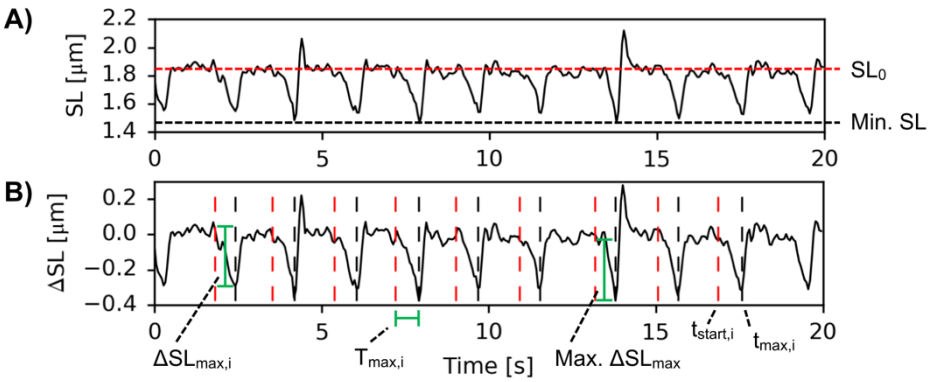

Figure 3-19: Definition of parameters of sarcomere contraction. (A) Sarcomere length, SL, vs. time. Red dashed line indicates the sarcomere length at rest, SL_0. Black dashed line indicates the minimal sarcomere length observed during full recording time, Min. SL. **(B)** Change in sarcomere length, ΔSL, vs. time. Red dashed lines indicate the start of contraction cycle i, $t_{start,i}$, black dashed lines indicate the time point of maximum contraction, $t_{max,i}$. $\Delta SL_{max,i}$ denotes the contraction amplitude of each single sarcomere contraction, respectively. $T_{max,i}$ denotes time-to-peak contraction. Max. $\Delta SL_{max,i}$ denotes the maximal contraction amplitude observed during full recording time.

To measure the level of coherence between the contractions of the individual sarcomeres within one myofibril, we turned to cross-correlation analysis. Briefly, for each contraction and each pair of sarcomeres, we analyzed the cross-correlation of ΔSL between the individual sarcomeres. To this end, each time interval of contraction was split up into ~ 20-30

smaller time intervals, and the pairwise cross-correlation was calculated for each interval. All these pairwise cross-correlations were then averaged, normalized and visualized in a cross-correlation color diagram (Figure 3-20-A). Here, the averaged pairwise cross-correlation for each time interval was visualized for each distance between sarcomeres, as measured in indices (Figure 3-20-B). The normalized cross-correlation was highest at maximum contraction, and to gain a measure for the level of coherence of contraction, we quantified the correlation at the time interval of maximum contraction ± 1 interval, averaged over these three intervals, and normalized by correlation at zero-index distance (Figure 3-20-C). Note, that in cross-correlation diagrams as displayed in Figure 3-20-A, we plotted time vs. inter-sacromeric distance, and colors indicate correlation in the respective time interval. High positive values (dark blue color) of the arbitrarily normalized correlation function indicate strong correlation. Values close to zero (white color) indicate no correlation, and high negative values (dark red color) indicate strong anti-correlation.

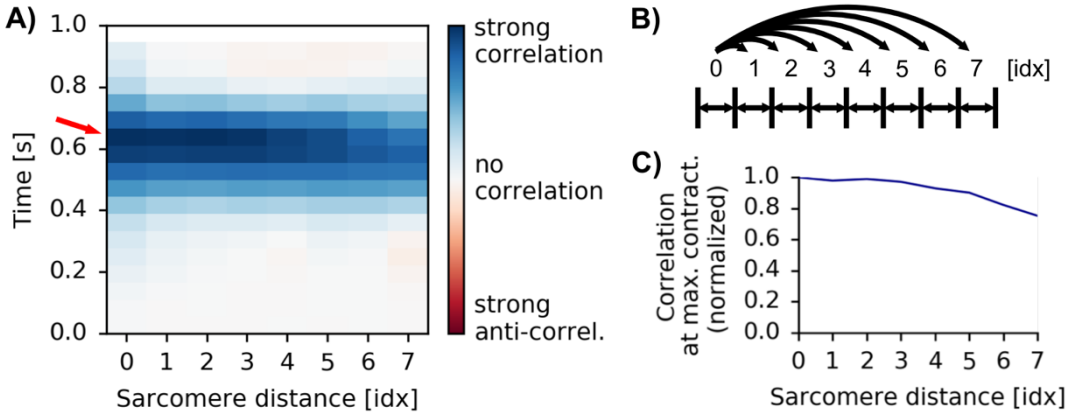

Figure 3-20: Cross-correlation color diagrams and correlation at maximum contraction. (A) Cross-correlation color diagram. At each time interval and for each contraction, the cross-correlation of ΔSL between all possible pairs of sarcomeres was calculated, normalized, and averaged over all contractions and pairs at distance i [idx]. Red arrow denotes time interval of maximum contraction. **(B)** Schematic depicting distance between sarcomeres, as measured in indices. **(C)** Cross-correlation at maximum contraction ± 1 time interval (normalized by correlation at zero-index distance) vs. sarcomere distance (see red arrow in (A)).

In a second approach to measure coherence between individual sarcomere contractions, we defined so-called *stereotypical contraction patterns* for each individual sarcomere. To this end, we superimposed all single contractions of each sarcomere, repectively, and averaged over all contractions to define stereotypical beating patterns. We then cross-correlated these stereotypical contraction patterns between the individual sarcomeres, and observed similar results as with non-stereotypical contractions (data not shown). However, it turned out that correlations analysis of stereotypical patterns was not as powerful in detecting subtle differences in coherence, as compared to the above-described correlation analysis of all individu-

al contractions. In this thesis, we thus refrained from evaluating results from correlation analysis of stereotypical patterns. However, we used stereotypical contraction patterns to create so-called *next-neighbor correlation diagrams* for further qualitative visualization of inter-sarcomeric coherence or incoherence (Figure 3-21).

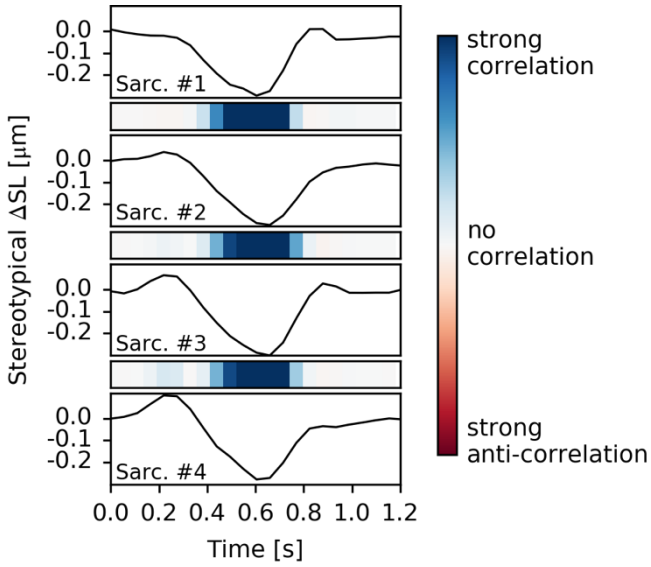

Figure 3-21: Next-neighbor correlation diagrams. Stereotypical contraction patterns for each individual sarcomere were created by superimposing all individual contractions of each sarcomere, respectively, and averaging over those contractions. Correlation diagrams indicate correlation of ΔSL between next-neighbor sarcomeres (i.e. idx = 1 in Figure 3-20-B). For illustration purposes only four sarcomeres are displayed.

3.3.2 The Effect of Substrate Elasticity on Sarcomere Contractility

To characterize the sarcomere dynamics of cardiomyocytes derived from the two transgenic cell lines, and to study the influence of ECM stiffness on sarcomere contractility, we ran the above-described contractility experiments with micropatterned CM from both lines seeded on either glass or 15 kPa elastomer substrates. In principle, it would further be revealing to investigate whether the insertion of the small linker peptide between ACTN2 and Citrine affected sarcomere dynamics by comparing the contractility of the two cell lines. However, we refrained from such comparison, as a) the quality of the micropatterns on 15 kPa may have been inconsistent (Section 3.1.3), and b) we observed relatively high fluctuations in the quality of differentiation of HES-2-ACTN2-Citr-CM in the early stages of this project, and all experiments presented in this thesis were conducted with only one batch of differentiation per each cell line. All experiments presented in this section were conducted in RPMI basal medium at 0.4 mM Ca^{2+}.

In the case of HES-2-Citr-Linker-derived CM, we evaluated contractility experiments with N = 10 cells on glass substrates and with N = 5 cells on 15 kPa elastomer substrates. For HES-2-Citr-4Mut-derived CM, N = 10 cells were evaluated on glass, and N = 11 cells were

analyzed on 15 kPa. Together, a total of 98 sarcomeres and a total of 159 beatings were analyzed in the case of HES-2-Citr-Linker-CM on glass substrates, corresponding to roughly 1560 individual sarcomere contractions. The according numbers for the other cell line/substrate combinations are displayed in Table 3-2. Analogously to Figure 3-18 (4Mut at 15 kPa), Figures 6-4 through 6-6 show representative kymographs, Z-line trajectories and ΔSL-vs.-time plots for the other cell line/substrate combinations.

Table 3-2: Number of experiments in the ECM rigidity assay.

Cell line/Substrate	# cells	# sarcomeres	# beatings	# sarc. contractions
Linker/Glass	10	98	159	1560
Linker/15 kPa	5	42	89	750
4Mut/Glass	10	118	109	1290
4Mut/15 kPa	11	135	124	1520

In the case of HES-2-Citr-Linker-CM on glass, we evaluated N = 10 cells, including a total of 98 sarcomeres and a total of 159 beatings, which corresponded to roughly 1560 individual sarcomere contractions. Numbers for further cell line/substrate combinations are displayed accordingly.

The average beating frequency of HES-2-Citr-Linker-CM was 0.84 ± 0.44 Hz on glass substrates, and 0.94 ± 0.39 Hz on 15 kPa (Figure 3-22-A). In the case of HES-2-Citr-4Mut-CM, it was 0.58 ± 0.15 Hz on glass, and 0.62 ± 0.12 Hz on 15 kPa. The differences between glass and 15 kPa substrates were not significant for either one of the cell lines ($p_{Linker/4Mut}$ = 0.68/0.53). To further evaluate, whether the variability of the time intervals between individual beatings was different for glass and 15 kPa, we analyzed the cell-specific relative standard deviation of the beating period, Rel. $STD_i(T)$, i.e. the standard deviation of the beating period (relative to the average period T_i) within each individual cell i, respectively (Figure 3-22-B). Again, no significant differences could be observed between glass and 15 kPa. For HES-2-Citr-Linker-CM, Rel. $STD_i(T)_{glass}$ was 6 ± 2% and Rel. $STD_i(T)_{15\ kPa}$ equaled 8 ± 5% (p = 0.20). For HES-2-Citr-4Mut-CM, Rel. $STD_i(T)_{glass}$ was 12 ± 5%, and Rel. $STD_i(T)_{15\ kPa}$ equaled 8 ± 5% (p = 0.07). These results suggested that ECM stiffness did not affect the consistency of the spontaneous beating period.

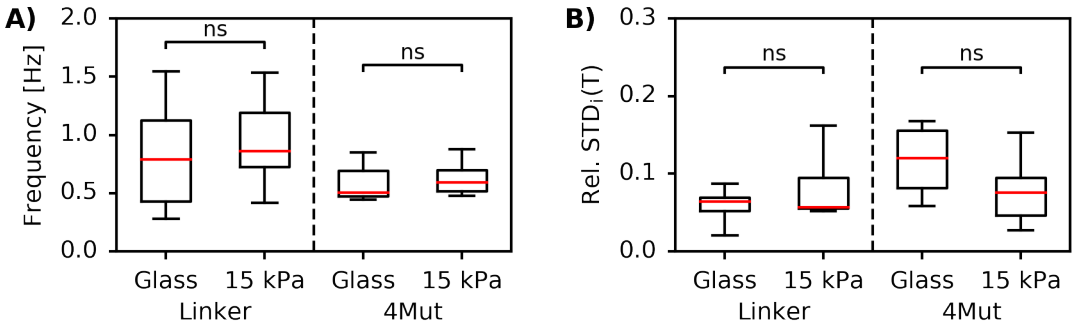

Figure 3-22: Beating frequency of HES-2-ACTN2-Citr-CM. (A) Beating frequency, f. For both cell lines, no significant differences could be observed between glass and 15 kPa. **(B)** Cell-specific relative standard deviation of the beating period between all individual contractions, Rel. $STD_i(T)$. No significant differences could be observed between glass and 15 kPa for both cell lines. $N_{Linker/Glass} = 10$, $N_{Linker/15\ kPa} = 5$, $N_{4Mut/Glass} = 10$, $N_{4Mut/15\ kPa} = 11$. ns: $p > 0.05$, *: $p < 0.05$; p-values as determined via unpaired Student's t-test.

Next, we wanted to evaluate sarcomere length and contraction amplitude of HES-2-ACTN2-Citr-CM. On average, the sarcomere length at rest, SL_0, was 1.83 ± 0.10 µm on glass, and 1.87 ± 0.13 µm on 15 kPa for HES-2-Citr-Linker-CM, and it was 1.72 ± 0.05 µm and 1.80 ± 0.07 µm for HES-2-Citr-4Mut-CM (Figure 3-23-A). While this difference was not significant in the case of HES-2-Citr-Linker ($p = 0.55$), it was significant for HES-2-Citr-4Mut ($p = 0.02$). For both cell lines, there was a clear but statistically insignificant trend towards shorter minimal sarcomere lengths, Min. SL, on 15 kPa (Figure 3-23-B). In the case of HES-2-Citr-Linker, Min. SL was 1.59 ± 0.12 µm on glass and 1.49 ± 0.13 µm on 15 kPa ($p = 0.18$), while it was 1.50 ± 0.05 µm and 1.46 ± 0.08 µm for HES-2-Citr-4Mut ($p = 0.12$). This trend for greater SL_0 but shorter Min. SL on 15 kPa suggested greater contraction amplitudes on 15 kPa substrates, as compared to glass. This notion was reinforced by the evaluation of the average of all individual contraction amplitudes, ΔSL_{max} (Figure 3-23-C). We observed ΔSL_{max} to be significantly greater by roughly 100% on the softer 15 kPa substrates in the case of both cell lines, where ΔSL_{max} was roughly 0.2 µm (see Table 3-3 and Table 3-4 for detailed data). Similar results were obtained by the analysis of the maximal contraction amplitude observed during recording time, Max. ΔSL_{max}, which was roughly twice as large as the average ΔSL_{max} in all cases (Figure 3-23-D, Table 3-3 and Table 3-4). As stated in Section 3.3.1, we chose to measure the parameters Min. SL and Max. ΔSL_{max} next to determining ΔSL_{max}, as we observed that the temporal resolution at 18.2 fps did not suffice to detect subtle changes in ΔSL_{max} in all of our experiments, especially in those where we studied the effects of drug treatment (see Section 3.4 and discussions in Sections 4.4, 4.6). See Figure 3-19 for details on the definition of the above parameters.

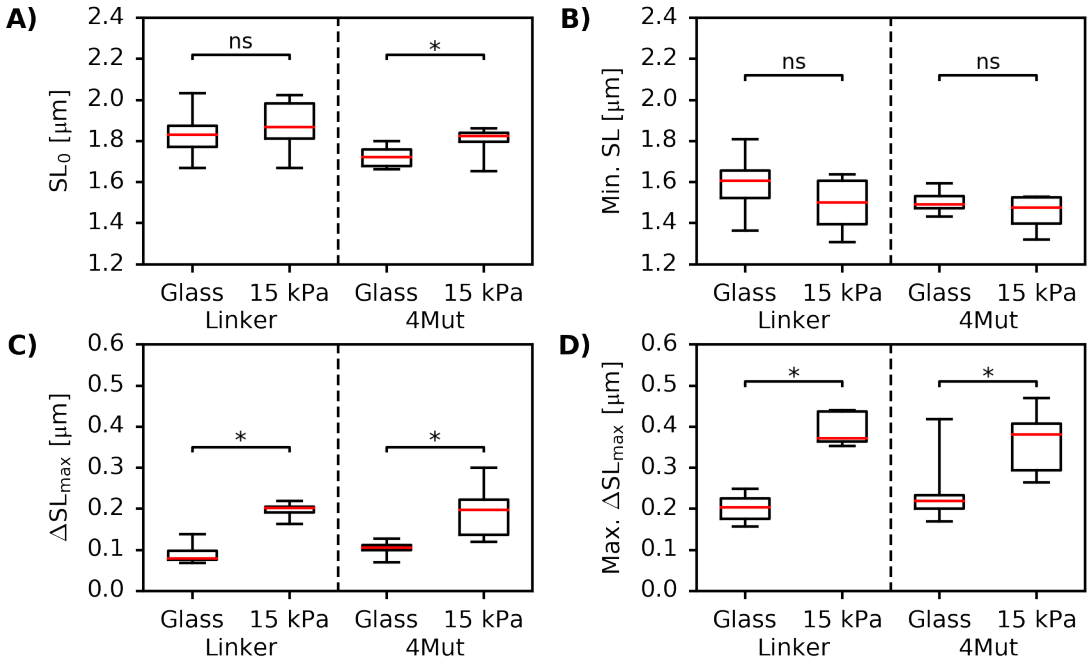

Figure 3-23: Sarcomere length and contraction amplitude of HES-2-ACTN2-Citr-CM. (A) Sarcomere length at rest, SL_0. **(B)** Minimal sarcomere length observed during recording time, Min. SL. **(C)** Average of individual sarcomere contraction amplitudes, ΔSL_{max}. **(D)** Maximal sarcomere contraction amplitude observed during recording time, Max. ΔSL_{max}. For details on the definition of all four parameters see Figure 3-19. $N_{Linker/Glass} = 10$, $N_{Linker/15\ kPa} = 5$, $N_{4Mut/Glass} = 10$, $N_{4Mut/15\ kPa} = 11$. ns: $p > 0.05$, *: $p < 0.05$; p-values as determined via unpaired Student's t-test.

Next, we evaluated the time-to-peak contraction, T_{max}, defined as the time interval between start of contraction and time point of maximum contraction (Figure 3-24-A). We observed no significant differences between glass or 15 kPa for either of the cell lines. T_{max} equaled 0.45 ± 0.13 s on glass and 0.39 ± 0.09 s on 15 kPa for HES-2-Citr-Linker-CM ($p = 0.38$), while it was 0.47 ± 0.09 s and 0.52 ± 0.09 s for HES-2-Citr-4Mut-CM ($p = 0.18$). To examine whether ECM rigidity affected the consistency of time-to-peak contraction, we further determined the cell-specific inter-sarcomeric standard deviation of T_{max} between individual contractions and relative to T_{max}, Rel. $STD_i(T_{max})$. It was significantly lower on 15 kPa in the case of both cell lines (Figure 3-24-B). Rel. $STD_i(T_{max})$ was 43 ± 11% on glass and 28 ± 8% on 15 kPa in the case of HES-2-Citr-Linker ($p = 0.02$), while it was 37 ± 9% and 25 ± 9% for HES-2-Citr-4Mut ($p < 0.01$). These results suggested that ECM stiffness influenced the fluctuation of contraction time between individual sarcomeres and/or individual beatings.

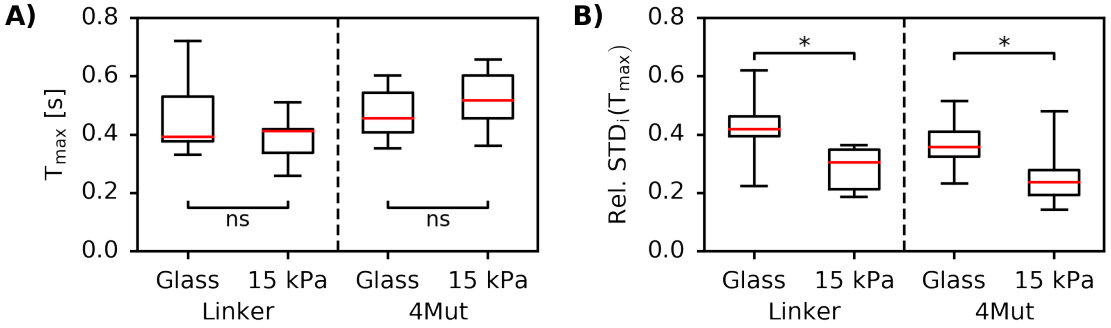

Figure 3-24: Time-to-peak contraction of HES-2-ACTN-Citr-CM. (A) Time-to-peak contraction, T_{max}. **(B)** Cell-specific, inter-sarcomeric standard deviation of T_{max}, Rel. $STD_i(T_{max})$. $N_{Linker/Glass} = 10$, $N_{Linker/15\ kPa} = 5$, $N_{4Mut/Glass} = 10$, $N_{4Mut/15\ kPa} = 11$. ns: $p > 0.05$, *: $p < 0.05$; p-values as determined via unpaired Student's t-test.

To gain insight into the level of coherence between individual sarcomere contractions within one myofibril, we examined the inter-sarcomeric cross-correlation of the change in sarcomere length at maximum contraction (Section 3.3.1). Figure 3-25 shows representative cross-correlation color diagrams for both, HES-2-Citr-Linker-CM and HES-2-Citr-4Mut-CM, on micropatterned glass and 15 kPa substrates. These diagrams correspond to the representative cells displayed in Figures 3-18 and 6-4 through 6-6.

The color diagrams in Figure 3-25 illustrate our general observation in contractility experiments, which suggested much greater levels of inter-sarcomeric coherence on 15 kPa substrates as compared to glass substrates. On glass, cross-correlation rapidly decreased to zero-correlation or even below, and thus indicated uncorrelated or even anti-correlated contractions at inter-sarcomeric distances as short as two sarcomeres (Figure 3-25-A/-C). In contrast, cross-correlation stayed high on 15 kPa even at inter-sarcomeric distances as high as seven sarcomeres and further[9] (Figure 3-25-B/-D).

[9] In all cells analyzed, we evaluated the motion of at least eight consecutive sarcomeres, and accordingly, our illustrations show data for inter-sarcomeric distances only up to 7 idx. When cells exhibited longer myofilaments, we extended our analysis to higher distances, and on 15 kPa, we consistently observed high levels of inter-sarcomeric coherence at least up to 12 idx.

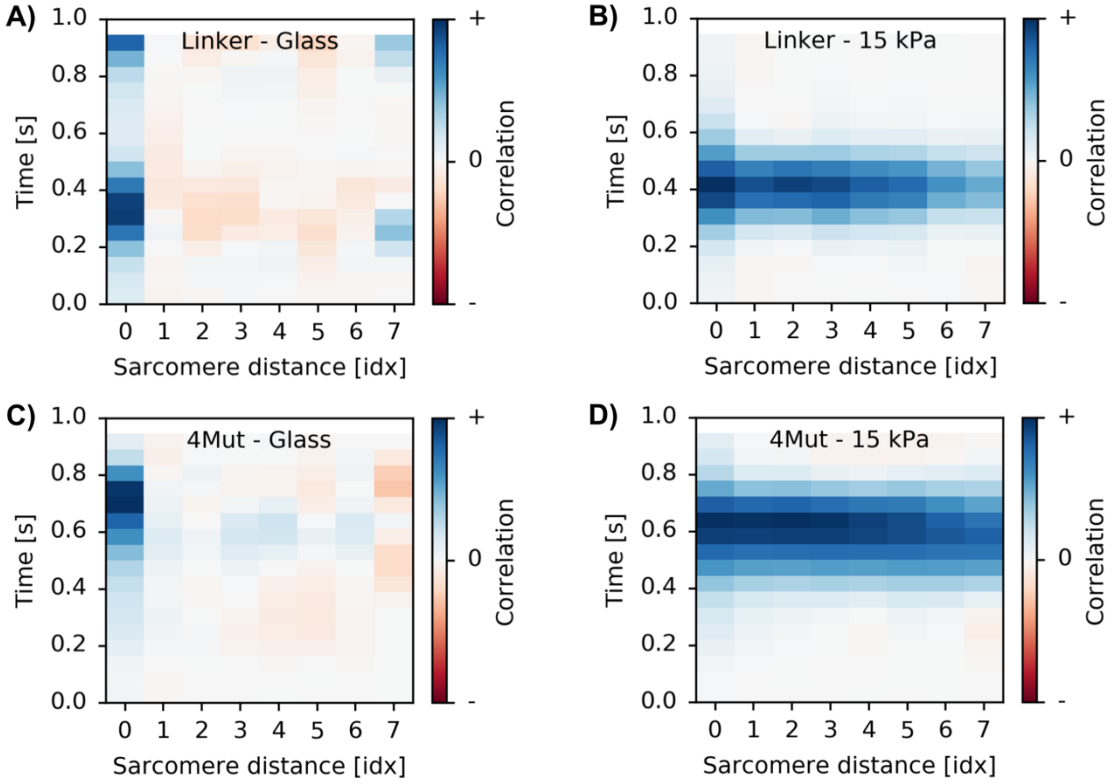

Figure 3-25: Cross-correlation color diagrams of HES-2-ACTN2-Citr-CM. (A) Representative diagram of a single HES-2-Citr-Linker-CM on a micropatterned glass substrate. **(B)** Representative diagram of a single HES-2-Citr-Linker-CM on micropatterned 15 kPa. **(C)** Representative diagram of a single HES-2-Citr-4Mut-CM on micropatterned glass. **(D)** Representative diagram of a single HES-2-Citr-4Mut-CM on micropatterned 15 kPa. These diagrams correspond to the representative cells displayed in Figures 3-18 and 6-4 through 6-6. See Figure 3-20 for details on cross-correlation color diagrams and sarcomere distance.

The normalized cross-correlation was highest at maximum contraction, and to gain a measure for the level of coherence of contractions, we quantified the correlation at the time interval of maximum contraction ± 1 interval, averaged over these three intervals, and normalized by correlation at zero-index distance (see Figure 3-20-B for further details). For both cell lines, Figure 3-26-A shows the average cross-correlation plotted against inter-sarcomeric distance for glass (blue) and 15 kPa substrates (red). The data mapped the above-stated general observation of greater inter-sarcomeric coherence on 15 kPa, and clearly indicated longer correlation lengths on these softer substrates. For further quantification of these observations, the inter-sarcomeric cross-correlation at distances of one, two, and three sarcomeres is displayed in Figure 3-26-B-D. On glass, cross-correlation at 1-, 2-, and 3-sarcomere distance was close to zero for both cell lines (see Tables 3-3 and 3-4 for detailed values), suggesting that contractions were highly uncorrelated between sarcomeres. On 15 kPa, cross-correlation at 1-, 2-, and 3-sarcomere distance was between 0.56 ± 0.33 and 0.72 ± 0.21 (Tables 3-3 and 3-4), indicating relatively strong correlation

between sarcomere contractions. Tables 3-3 and 3-4 summarize all above-presented data for both cell lines, respectively.

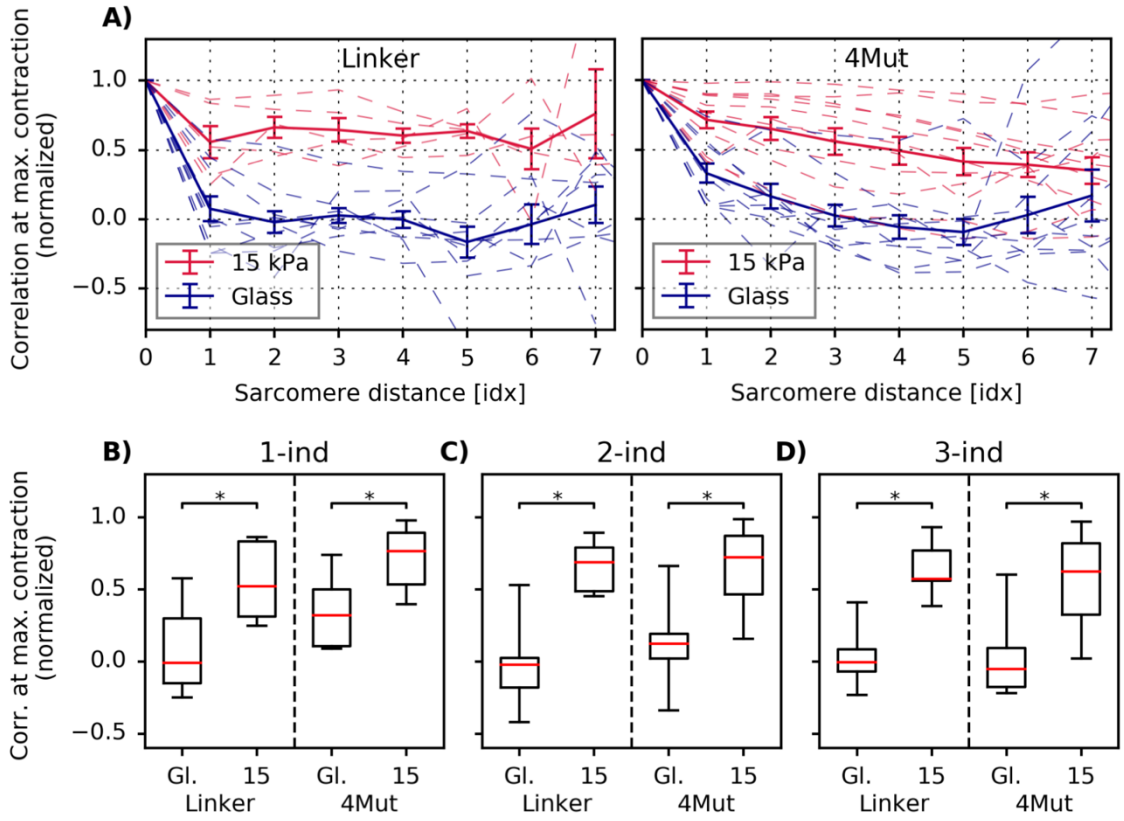

Figure 3-26: Average inter-sarcomeric cross-correlation of HES-2-ACTN2-Citr-CM. **(A)** Inter-sarcomeric cross-correlation of the change in sarcomere length at maximum contraction (calculated as described in Section 3.3.1) vs. inter-sarcomeric distance. Dashed lines indicate results from individual contractility experiments (i.e. individual cells). Solid lines mark the corresponding mean values. Here, error bars mark one standard error of the mean (SEM). **(B)-(D)** Inter-sarcomeric cross-correlation at inter-sarcomeric distances 1 idx, 2 idx, and 3 idx, respectively. Here, error bars mark one standard deviation. See Figure 3-20 for details on sarcomere distance. Gl.: glass, 15: 15 kPa. $N_{Linker/Glass} = 10$, $N_{Linker/15\ kPa} = 5$, $N_{4Mut/Glass} = 10$, $N_{4Mut/15\ kPa} = 11$. ns: $p > 0.05$, *: $p < 0.05$; p-values as determined via unpaired Student's t-test.

Table 3-3: Parameters of contractility of HES-2-Citr-Linker-CM.

	Glass (N = 10)	15 kPa (N = 5)	p-value
f [Hz]	0.84 ± 0.44	0.94 ± 0.39	0.68
Rel. $STD_i(T)$ [%]	6.1 ± 1.9	8.4 ± 4.3	0.20
SL_0 [µm]	1.83 ± 0.10	1.87 ± 0.13	0.55
Min. SL [µm]	1.59 ± 0.12	1.49 ± 0.13	0.18
ΔSL_{max} [µm]	0.09 ± 0.03	0.20 ± 0.02	**< 0.001**
Max. ΔSL_{max} [µm]	0.20 ± 0.04	0.39 ± 0.04	**< 0.001**
T_{max} [s]	0.45 ± 0.13	0.39 ± 0.09	0.38
Rel. $STD_i(T_{max})$ [%]	43.1 ± 10.1	28.4 ± 7.2	**0.02**
Ind-1 correlation	0.07 ± 0.28	0.56 ± 0.26	**< 0.01**
Ind-2 correlation	-0.02 ± 0.25	0.66 ± 0.18	**< 0.001**
Ind-3 correlation	0.03 ± 0.18	0.64 ± 0.19	**< 0.001**

On glass (15 kPa), 10 (5) cells, including a total of 98 (42) sarcomeres and a total of 159 (89) beatings (corresponding to roughly 1560 (750) individual sarcomere contractions), were analyzed. Significant p-values, as determined via Student's unpaired t-test, were highlighted.

Table 3-4: Parameters of contractility of HES-2-Citr-4Mut-CM.

	Glass (N = 10)	15 kPa (N = 11)	p-value
f [Hz]	0.58 ± 0.15	0.62 ± 0.12	0.53
Rel. $STD_i(T)$ [%]	11.6 ± 4.2	8.0 ± 4.1	0.07
SL_0 [µm]	1.72 ± 0.05	1.80 ± 0.07	**0.02**
Min. SL [µm]	1.50 ± 0.05	1.46 ± 0.08	0.12
ΔSL_{max} [µm]	0.10 ± 0.02	0.19 ± 0.06	**< 0.001**
Max. ΔSL_{max} [µm]	0.23 ± 0.07	0.36 ± 0.07	**< 0.001**
T_{max} [s]	0.47 ± 0.09	0.52 ± 0.09	0.18
Rel. $STD_i(T_{max})$ [%]	37.0 ± 8.6	25.1 ± 9.0	**< 0.01**
Ind-1 correlation	0.33 ± 0.22	0.72 ± 0.21	**< 0.001**
Ind-2 correlation	0.16 ± 0.29	0.65 ± 0.28	**< 0.01**
Ind-3 correlation	0.02 ± 0.26	0.56 ± 0.33	**< 0.001**

On glass (15 kPa), 10 (11) cells, including a total of 118 (135) sarcomeres and a total of 109 (124) beatings (corresponding to roughly 1290 (1520) individual sarcomere contractions), were analyzed. Significant p-values, as determined via Student's unpaired t-test, were highlighted.

3.3.3 Ca^{2+}-imaging in HES-2-ACTN2-Citr-derived Cardiomyocytes

We performed proof-of-concept contractility experiments with simultaneous Ca^{2+}-imaging in HES-2-ACTN2-Citr-CM. Ca^{2+}-imaging was performed at room temperature in RPMI basal medium at 0.4 mM Ca^{2+} using the membrane-permeant calcium indicator dye Fluo-4 AM (Cat F14201, Thermo Fisher Scientific, Waltham, USA).

Figure 3-27-A shows a single frame from a typical contractility experiment with simultaneous Ca^{2+}-imaging. The mean fluorescence intensity, as measured over time for ROI 1, 2, and 3, respectively, indicated periodic cytosolic calcium increase (Figure 3-27-B). In Figure 3-27-C, the intensity of all three ROIs was averaged and plotted next to the change in sarcomere length of a single representative sarcomere, as determined via our analysis algorithm (note that the recording of calcium intensity for an entire 20 seconds and the respective Z-line trajectories of all eight Z-lines marked by the pink line in Figure 3-27-A are displayed in Figure 6-7). Start of sarcomere contractions, as indicated by the red dashed lines, occurred at the steep increase of intracellular calcium concentration, as expected for cardiac contractions (Bers 2002).

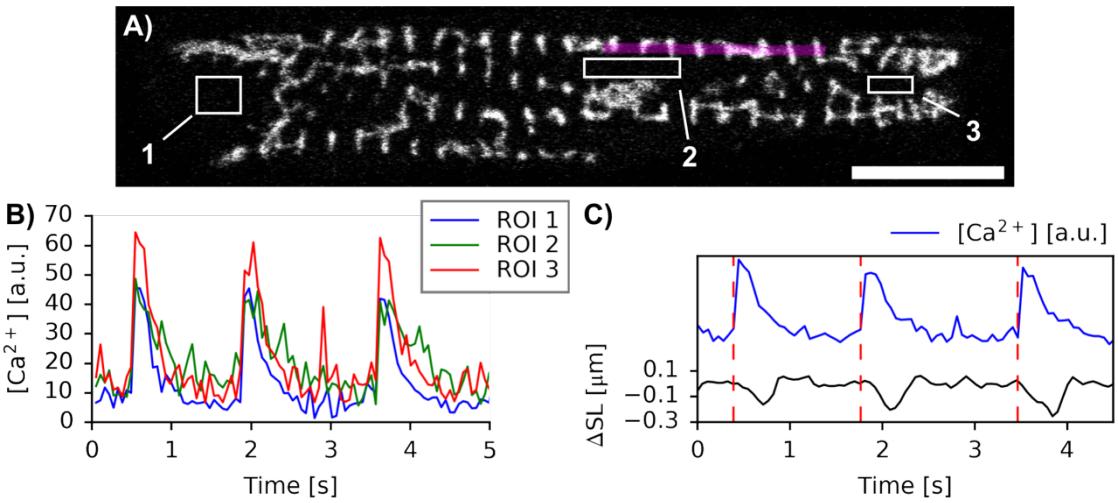

Figure 3-27: Calcium imaging in HES-2-ACTN2-Citr-CM. (A) Single HES-2-Citr-Linker-CM on a micropatterned glass substrate. White boxes mark ROIs where fluorescence intensity flux from calcium dye was measured. The cell was imaged for 35 seconds at 18.2 fps, and the mean intensity for each ROI, respectively, was recorded. **(B)** Calcium concentration over time, as derived from the fluorescence intensity flux in the three ROIs in panel A. Only five seconds shown for illustration purposes. **(C)** Change in sarcomere length for one representative sarcomere (black) and calcium intensity (blue) over time. Dashed lines indicate start of contraction, as derived via our analysis algorithm (Section 3.3.1). Only three beatings displayed for illustration purposes. The recording of calcium intensity for an entire 20 seconds and the respective Z-line trajectories of all eight Z-lines marked by the pink highlighting in Figure 3-27-A are displayed in Figure 6-7. Scale bar 10 µm.

3.4 The Effect of Omecamtiv Mecarbil on Sarcomere Contractility

To show proof of concept that our indicator system can function as a versatile tool in drug screening assays, we conducted contractility experiments in the presence of the selective cardiac myosin (MYH7) activator omecamtiv mecarbil (OM). OM is currently in phase-3 clinical trials for its application in the treatment of heart failure and has been reported to improve cardiac function in patients with systolic heart failure (Cleland et al. 2011; Teerlink et al. 2011; Teerlink et al. 2016; Teerlink et al. 2020). It has created large excitement, as cardiac myosin activators may avoid potential side effects that limit the use of other pharmaceuticals that are being used in the treatment of heart failure, since these typically target upstream regulatory mechanisms of cardiac beating (Ponikowski et al. 2016; Woody et al. 2018). When first introduced, OM was thought to directly activate myosin via an allosteric mechanism that increases the transition rate of myosin into the strongly actin-bound force-generating state (Morgan et al. 2010; Malik et al. 2011), however, the results from multiple subsequent studies have led to question this direct mechanism on a structural, biochemical and physiological basis (for an overview, refer to Woody et al. (2018)). In a latest biophysical approach exploiting optical trapping, Woody et al. (2018) proposed that OM's mechanism of action was, in fact, not direct, but that it *i)* stabilizes the ADP-P pre-power stroke state and increases the entry rate into the state of strong actin binding through increase of ADP-P phosphate release rates, as previously stated by Malik et al. (2011), and it *ii)* prolongs the strong actin-bound state by inhibiting myosin molecules, therefore boosting thin-filament activation. Interestingly, Woody et al. (2018) further stated that OM *i)* reduces the myosin work stroke, *ii)* renders the actin-detachment rate independent of both, applied load and ATP concentration, *iii)* inhibits the velocity of actin gliding, and *iv)* at micromolar (i.e. high) concentrations, reduces the rate of tension development and relaxation in CM.

We sought to gain further insight into the effects of OM on CM beating via testing the direct and dynamic morphological feedback of HES-2-ACTN2-Citr-CM in contractility experiments in the presence of OM. The general experimental sequence was as follows: First, CM were video imaged at zero-drug concentration. Then, the drug was pipetted and mixed into the medium at subsequently increasing concentrations. The subsequent video recordings were started five minutes after adding the indicated amount of pharmaceutical to the solution. All experiments presented in this section were performed with CM derived from HES-2-Citr-4Mut.

On 15 kPa, we performed contractility experiments at OM concentrations ranging from 10 nM to 10 μM, as well as at zero-concentration for control. Only in the case of six cells, data at different concentrations were obtained for each same cell, respectively, and only these six cells were included in the following analysis. To account for the inter-cellular variance in characteristic parameters of CM beating among a relatively small number of analyzed cells, we chose to display individual line graphs for each cell, respectively, next to

concentration-response graphs indicating the mean ± one standard deviation. All OM contractility experiments on 15 kPa were conducted in RPMI basal medium at 0.4 mM Ca^{2+}.

Figure 3-28 shows kymographs of four representative Z-lines in a single representative cardiomyocyte at five different OM concentrations. A first qualitative analysis of videos and/or kymographs showed homogeneous, periodic beating up to 100 nM OM among an increase in beating rate and an increase in the amplitude of Z-line displacement as a sign of hypercontraction. At 1 μM OM, sarcomeric beating seemingly became inhomogeneous, and at 10 μM OM, most of the sarcomeres entered into what appeared to be a state of constant contracture.

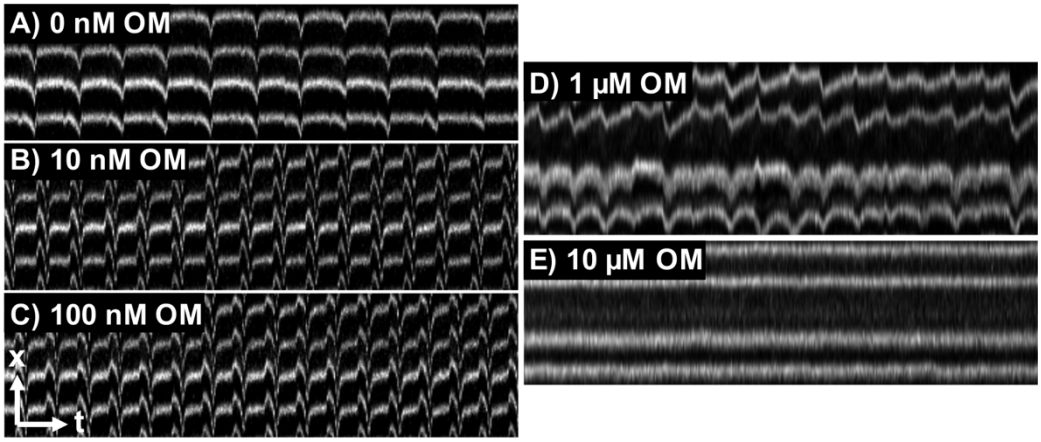

Figure 3-28: Kymographs of a representative HES-2-Citr-4Mut-CM treated with OM. (A)-(E) Kymographs for four representative Z-lines of a single representative HES-2-Citr-4Mut-CM on 15 kPa at different concentrations of omecamtiv mecarbil, as noted.

The unexpected qualitative notion of a positive chronotropic effect of OM on CM beating could be corroborated quantitatively (Figure 3-29). Treatment with OM resulted in an increase in frequency to about 150% baseline at micromolar concentrations ($f_{0\,nM}$ = 0.56 ± 0.05 Hz, $f_{10\,\mu M}$ = 0.84 ± 0.03 Hz, $p < 0.001$, n = 6 cells, see Table 3-5 for further data points). To further support this observation, Figure 6-8 shows the concentration-frequency relationship obtained from a separate contractility assay, which we recorded on glass substrates in the presence of OM at 1 nM to 100 μM[10]. These recordings on glass reinstated the notion of a positive chronotropic effect.

[10] Contractility experiments on glass substrates were performed in DMEM at 0.8 mM Ca^{2+}. In principal, we here observed trends similar to those on 15 kPa. However, since we only recorded one cell on glass, and since the full development of sarcomere dynamics seemed to be greatly hindered by the stiff environment, which resulted in greater noise induced from pixel resolution (see Section 3.3.2), we here refrained from presenting all data analogously to those for 15 kPa.

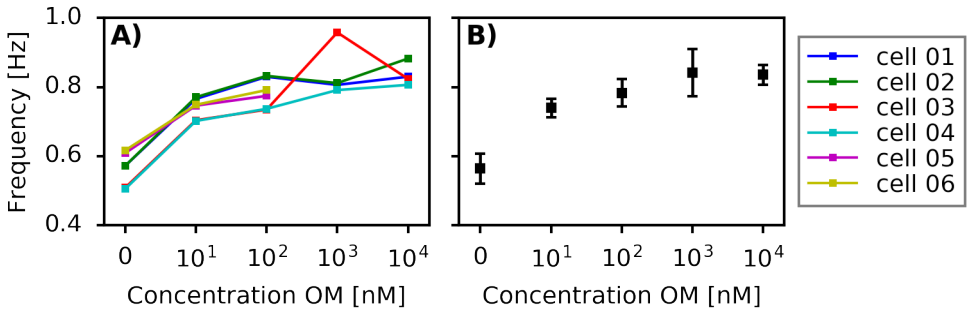

Figure 3-29: The effect of OM on beating frequency. (A) Concentration-frequency curves for each cell, individually, as determined for HES-2-Citr-4Mut on 15 kPa. **(B)** Concentration-frequency relationship illustrated as average ± one standard deviation. In the case of 10 µM OM, where most sarcomeres had entered into a state of constant contracture, the "beating" frequency was estimated from an overall undulation of the cells' Z-lines (see Figure 3-28 and Figure 6-9). At all other OM concentrations, the frequency was obtained via our analysis algorithm. A modeling of the (presumably sigmoidal) concentration-response relationship was not applicable since there were no data available at relatively low (pico- to low nanomolar) OM concentrations. $p < 0.001$, as determined via rmANOVA ($n = 6$). To further support the unexpected positive chronotropic effect of OM on CM beating, Figure 6-8 shows the concentration-frequency relationship obtained from a separate contractility assay, which we recorded on a glass in the presence of OM at 1 nM to 100 µM.

Next, we wanted to evaluate the effect of OM on sarcomere shortening amplitude. The minimal observed sarcomere length, Min. SL, became significantly shorter with increasing OM concentration (Figure 3-30-A). At 100 nM OM, Min. SL was shorter by 4.3 ± 2.9%, as compared to 0 nM OM, and at 10 µM OM, Min. SL was shorter by 14.9 ± 4.3% ($p < 0.001$). This observed decrease in Min. SL suggested an expected positive inotropic effect of OM and was in line with the above-described qualitative notion of hypercontraction.

Despite these signs of hypercontraction, we did not observe a significant increase in our standard parameter for contraction amplitude, i.e. the average of the individual contraction amplitudes, ΔSL_{max}, nor in its maximal value, Max. ΔSL_{max} (Figure 3-30-B). We observed $\Delta SL_{max,0\ nM} = 0.25 \pm 0.04$ µm, $\Delta SL_{max,10\ nM} = 0.22 \pm 0.02$ µm, and $\Delta SL_{max,100\ nM} = 0.24 \pm 0.02$ µm ($p = 0.15$, see Table 3-5 for analogous values for Max. ΔSL_{max}). The lack of a significant increase in ΔSL_{max} with increasing [OM] could most likely be attributed to the limited temporal resolution (18.2 fps), that seemed not to have sufficed to detect the subtle differences in contraction amplitude between different OM concentrations (see Sections 4.6 and 4.7 for detailed discussions).

Notably, there was a significant trend towards decreasing sarcomere rest lengths (SL_0) with increasing OM concentration ($p < 0.001$, Figure 3-30-A). The addition of 100 nM OM

resulted in a decrease in sarcomere rest length by 2.6 ± 1.5% on average, as compared to 0 nM control. At micromolar concentrations of OM, the average observed SL_0 seemed to increase again. At these high concentrations, some few sarcomeres seemed to get pulled apart by the (presumably stronger) neighboring sarcomeres, which became particularly evident in the state of constant contracture at 10 µM (see Figure 6-9 and kymograph in Figure 3-28-E). Our analysis thus yielded larger average rest lenghts for all sacormeres, despite even shorter SL_0 for the non-disrupted sarcomeres. To account for this bias, only concentrations 0 to 100 nM were included in the analysis of SL_0.

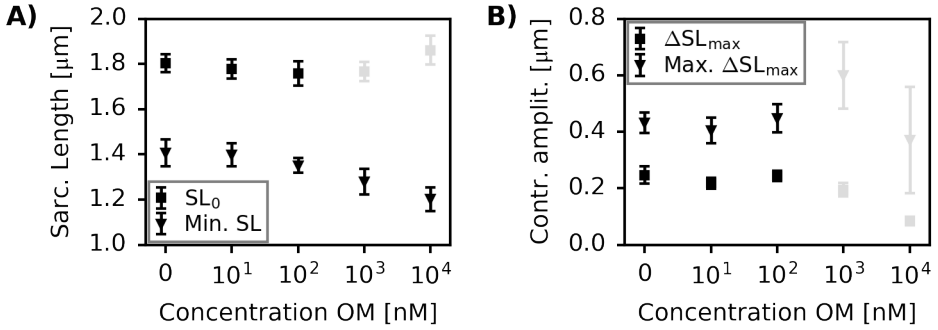

Figure 3-30: Effects of OM on sarcomere length and contraction amplitude. Concentration-response relationships were determined on 15 kPa. **(A)** Sarcomere length at rest (SL_0, $p < 0.001$) and minimal sarcomere length observed during recording time (Min. SL, $p < 0.001$). Micromolar concentrations were excluded from analysis of SL_0 because ruptured sarcomeres resulted in biased average rest lengths (see text for details). **(B)** Average contraction amplitude (ΔSL_{max}, $p = 0.15$) and maximal contraction amplitude observed during recording time (Max. ΔSL_{max}, $p = 0.29$). Micromolar concentrations were excluded from the analysis as the video analysis algorithm was not applicable to determine ΔSL_{max} for highly uncoordinated beatings at 1 µM and at the state of constant contracture at 10 µM OM. p-values as determined via rmANOVA. N = 6 cells for 0 to 100 nM OM. N = 4 cells for 1 to 10 µM OM. Figure 6-10 shows the corresponding individual line graphs for each cell, individually.

We next evaluated the time-to-peak contraction (T_{max}), and our analysis suggested a tendency towards shorter T_{max} with increasing concentration of OM (Figure 3-31, Table 3-5). We observed $T_{max,0\ nM} = 0.62 \pm 0.03$ s and $T_{max,100\ nM} = 0.58 \pm 0.06$ s ($p = 0.03$), however, a higher number of experiments would have been required to allow for a relevant statement about whether OM affected time-to-peak contraction. Again, at micromolar concentrations, our analysis algorithm did not apply to determine time points of starting and peak contraction, and these data were excluded from analysis. All above-stated results are summarized in Table 3-5.

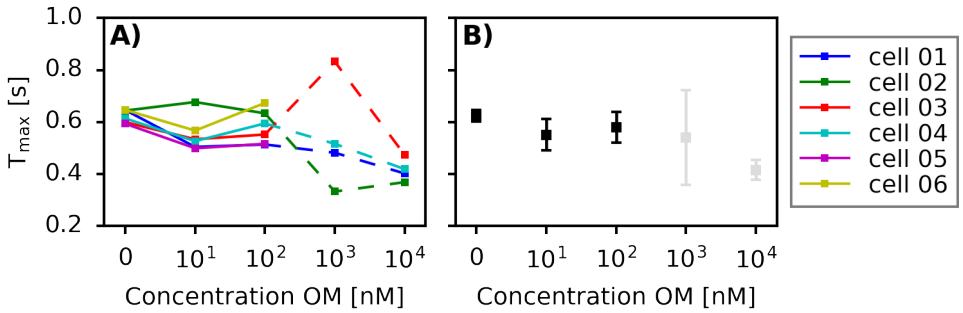

Figure 3-31: The effect of OM on time-to-peak contraction. (A) Concentration-T_{max} curves for each cell, individually, as determined on 15 kPa. **(B)** Concentration-T_{max} relationship illustrated as average ± one standard deviation. p = 0.03, as determined via rmANOVA. For uncoordinated contractions at micromolar concentrations, our analysis algorithm did not apply to determine time points of starting and peak contraction, and thus T_{max}, such that these data were excluded from the analysis.

Table 3-5: Effects of omecamtiv mecarbil on sarcomere contractility.

Conc. OM	0 nM (N = 6)	10 nM (N = 6)	100 nM (N = 6)	1 µM (N = 4)	10 µM (N = 4)
f [Hz] *	0.56 ± 0.05	0.74 ± 0.03	0.78 ± 0.04	0.84 ± 0.07	0.84 ± 0.03
SL_0 [µm] *	1.80 ± 0.05	1.78 ± 0.05	1.76 ± 0.06	(1.77 ± 0.05)	(1.86 ± 0.07)
Min. SL [µm] *	1.41 ± 0.06	1.40 ± 0.06	1.35 ± 0.04	1.28 ± 0.06	1.20 ± 0.06
ΔSL_{max} [µm]	0.25 ± 0.04	0.22 ± 0.02	0.24 ± 0.02		
Max. ΔSL_{max} [µm]	0.43 ± 0.04	0.41 ± 0.05	0.45 ± 0.06		
T_{max} [s] *	0.62 ± 0.03	0.55 ± 0.07	0.58 ± 0.06		
Ind-1 correl. *	0.77 ± 0.09	0.66 ± 0.12	0.67 ± 0.18	0.11 ± 0.13	- 0.36 ± 0.17

At micromolar concentrations, values for SL_0 were biased by disrupted sarcomeres. Further, our video analysis algorithm was not applicable to determine ΔSL_{max} and T_{max} for highly uncoordinated beatings at 1 µM and at the state of constant contracture at 10 µM OM. Ind1-correlation denotes next-neighbor correlation of ΔSL at maximum contraction (see below). *: $p < 0.05$, as determined via rmANOVA.

We further sought to evaluate potential effects of omecamtiv mecarbil on the intersarcomeric coherence of contractions. Figure 3-32 displays the next-neighbor correlation of the change in sarcomere length at maximum contraction for different OM concentrations. Although our video analysis algorithm was not applicable to determine ΔSL_{max} at ≥ 1 µM OM (see above), it was able to determine ΔSL(t) and therefore its correlation between sarcomeres, such that we were able to include all recorded concentrations of OM in the analysis.

We observed a significant tendency towards lower correlation with increasing concentration of OM ($p < 0.001$). A higher number of experiments would have been required to further support the observed trend in the nanomolar concentration range, where post-hoc paired t-tests yielded a significant decrease between 0 nM and 10 nM OM ($p = 0.04$), but no significant decrease between 0 nM and 100 nM OM ($p = 0.16$). However, the clear drop of correlation close to zero at 1 µM OM (0.11 ± 0.13) was evident, and it was in line with the qualitative observation that sarcomere contractions became highly inhomogeneous, i.e. uncorrelated. At 10 µM OM, correlation dropped even further to -0.36 ± 0.17, thus indicating a certain amount of anti-correlation between sarcomere contractions. This observation was consistent with the qualitative notion, that at 10 µM OM, cardiomyocytes entered into a state of constant contracture, where some few single contractions of single sarcomeres occurred very sparsely. If part of a myofilament did contract, some single sarcomeres in other parts of the myofilament seemed to get pulled apart, and thus the change in sarcomere length ΔSL between these sarcomeres was anti-correlated. All ind1-correlation coefficients are listed in Table 3-5.

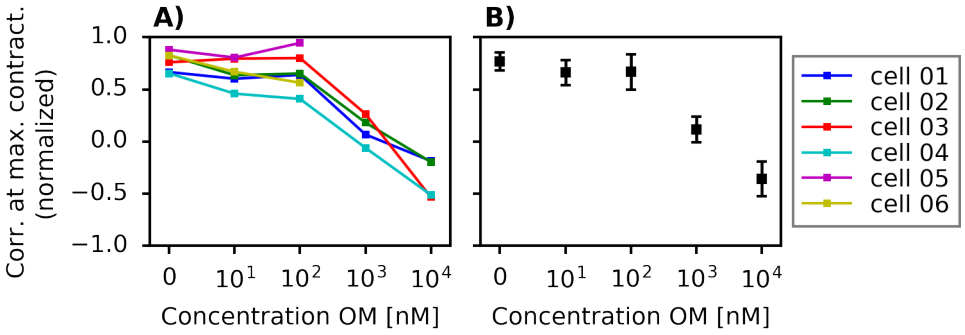

Figure 3-32: The effect of OM on next-neighbor correlation. Concentration-effect relationship for next-neighbor correlation at maximum contraction was determined on 15 kPa. **(A)** Concentration-correlation curves for each cell, individually. **(B)** Concentration-correlation relationship illustrated as average ± one standard deviation. $p < 0.001$, as determined via rmANOVA. While our video analysis algorithm was not applicable to reliably determine ΔSL_{max} at ≥ 1 µM OM, it was able to measure ΔSL(t) and therefore its correlation.

3.5 Temporal Development of the Sarcomeric Network

Endogenous fluorescent labeling of Z-lines raised the intriguing prospects to visualize the temporal development of sarcomeric myofibrils in live CM. In this section, we show proof of concept that our newly developed knock-in lines facilitate studies on the dynamic temporal development of the sarcomeric network in live HES-2-derived cardiomyocytes.

3.5.1 Time Scales of Sarcomere Assembly

To study time scales of myofibrillogenesis, we digested HES-2-ACTN2-Citr-CM using Accutase®, and subsequently seeded the cells onto Synthemax®-coated substrates. We then recorded confocal z-stack time series of the developing cardiac myocytes.

In Figure 3-33, a single representative CM, here derived from HES-2-Citr-Linker, is displayed at various time points after seeding. In this particular cell, clear formation of sarcomeric myofibrils was observed within two hours after seeding, and periodic beating became apparent after roughly 2.5 hours. In other experiments, the formation of well-assembled, contracting myofibrils could be observed within < 30 minutes after seeding (data not shown). These observations demonstrate the applicability of our knock-in lines to the dynamic studying of myofibrillogenesis in live CM, however, a higher number of experiments at smaller time steps need to be performed to allow for more qualitative and quantitative statements regarding the formation of the sarcomeric network.

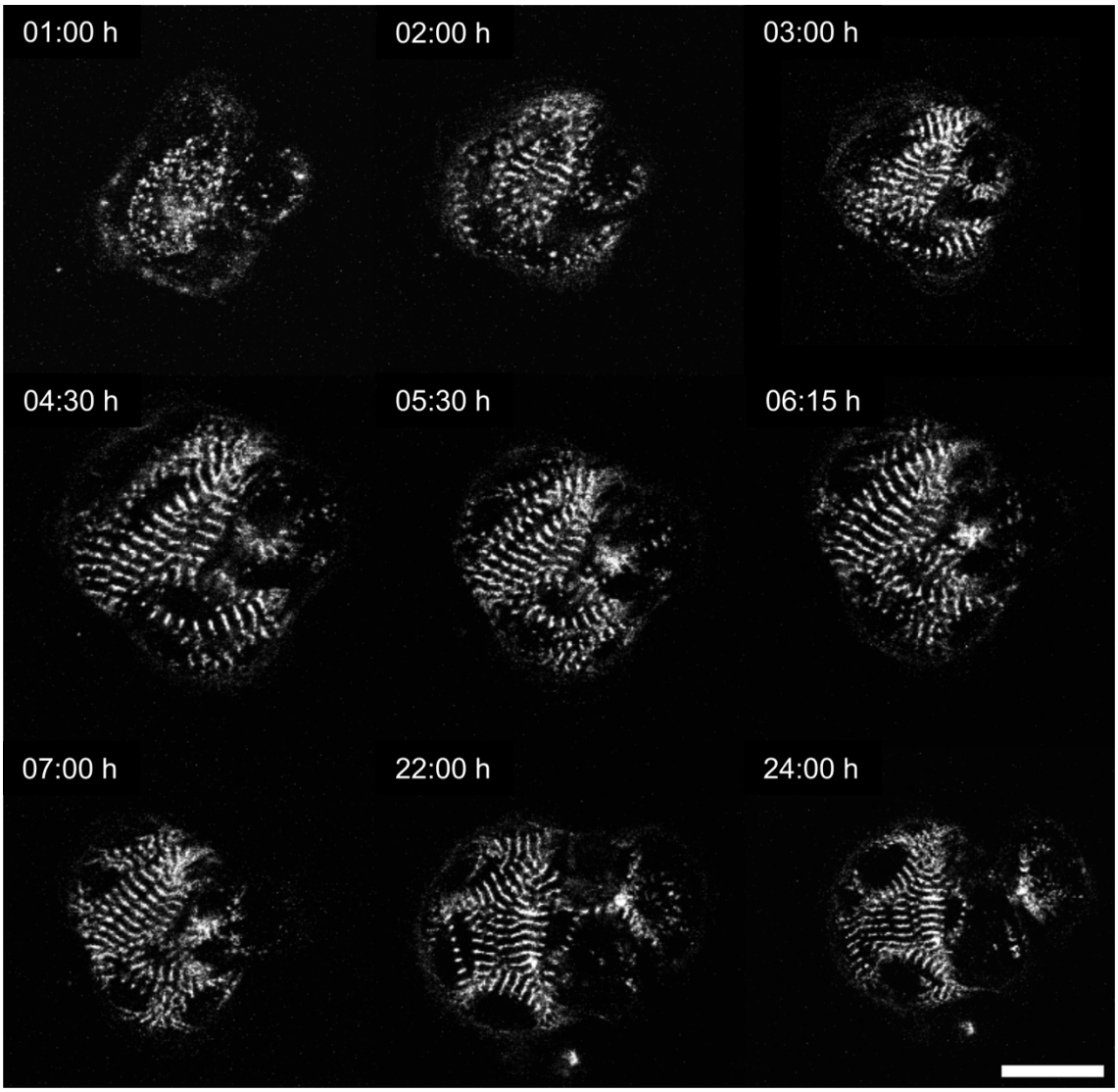

Figure 3-33: Temporal development of sarcomeres. Cardiomyocytes, here derived from HES-2-Citr-Linker, were digested with Accutase® and seeded on Synthemax®-coated glass coverslips. Z-stack time series of the developing cells were recorded; here only slices at the coverslip plane are shown. First sarcomeric beating of this particular cell was observed at roughly 2.5 h after seeding. Scale bar 20 μm (only applies to images in third row).

3.5.2 Disassembly of Myofibrils During Digestion with Accutase®

To gain insight not only into the assembly of cardiac sarcomeric networks, but also into the disassembly of myofibrils, we confocally imaged live HES-2-ACTN2-Citr-CM during the digestion process with Accutase® at one-minute time intervals (Figure 3-34). To this end, we seeded HES-2-ACTN2-Citr-CM on micropatterned 15 kPa elastomer substrates, and after five days of cell culture, we transferred the substrates to our confocal microscope setup before starting the digestion process according to our standard protocol (Section 6.4). Digestion was performed at room temperature and 5% CO_2.

Figure 3-34 shows several time frames from a typical digestion experiment. To mimic the 3-dimensional aspect of the digestion process, the frames display the z-projection over 36 z-slices at 0.5 µm z-step size (slices summed). First signs of cell detachment were observed roughly 15 minutes after start of digestion, as indicated by the white arrows. Throughout all our digestion experiments (N = 4 cells), detachment of CM first started at the far end of the myofibrils, and the cells would then roll together as the detachment process proceeded. Notably, the sarcomeres appeared to stay morphologically intact and connected even in the detached areas of the cells, such that the formerly elongated myofibrils were still visible at the cell boundaries (clearly visible only in a 3D reconstruction of the z-stack time series).

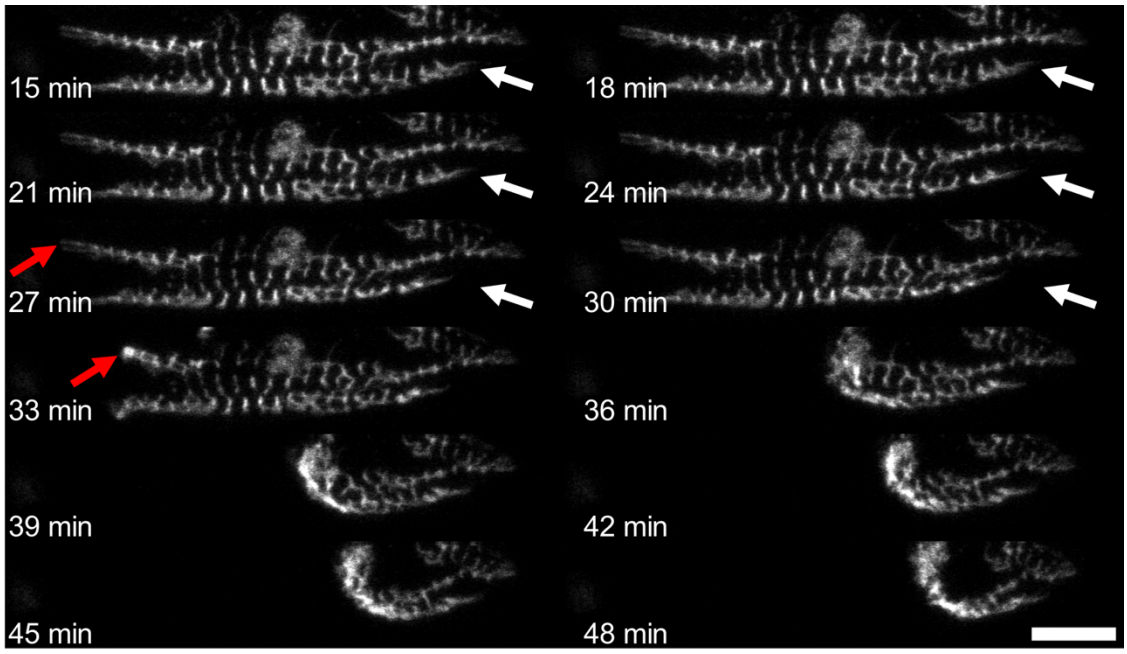

Figure 3-34: Disassembly of myofibrils during Accutase® digestion. CM, here derived from HES-2-Citr-Linker, were seeded on micropatterned 15 kPa elastomer substrates, and the detachment of the cells during the digestion with Accutase® was imaged confocally at one-minute time steps (here, only images at three-minute time intervals are shown). To mimic the 3-dimensional aspect of the digestion process, the frames display the z-projection over 36 z-slices at 0.5 µm z-step size (slices summed). First signs of cell detachment were observed roughly 15 minutes after start of digestion, as indicated by the white arrows. In all digestion experiments (N = 4 cells), detachment of the cells was first observed at the far end of the myofibrils (see red arrows), and the cells would then roll together before becoming entirely unattached. Scale bar 10 µm.

4 Discussion

The aim of this thesis was to develop a robust fluorescence indicator system for the *in-vitro* characterization of human stem cell-derived cardiomyocytes. Part of the indicator system was a novel cell line that was designed to express a stable fluorescent label of one of the sarcomeric proteins, and as such to allow for longitudinal studies which should a) help to understand basic mechanisms of sarcomere dynamics and synchronization of hPSC-CM, and b) prove useful to the potential use of hPSC-CM in high-throughput drug screening assays. Pursuing this aim, we chose to combine CRISPR/Cas9-mediated endogenous fluorescent tagging of sarcomeric α-actinin (discussed in Section 4.1) with efficient micropatterning on substrates of different elasticity (Section 4.2). By custom developing a powerful video analysis algorithm and applying it to contractility assays of micropatterned HES-2-ACTN2-Citr-derived cardiomyocytes, we

A) were able to analyze sarcomere dynamics at the subcellular scale (Section 4.3),
B) gained insight into the mechanical crosstalk between the contractile elements of cardiomyocytes (Section 4.3),
C) gained insight into how this crosstalk is affected by the stiffness of the surrounding environment (Section 4.3), and
D) demonstrated proof-of-concept applicability of our indicator model in drug testing (Section 4.4).
E) Further, we showed applicability of our indicator model in the dynamic studying of myofibrillogenesis (Section 4.5).

4.1 CRISPR/Cas9-mediated Fluorescent Labeling of Z-lines in Cardiomyocytes

Employing CRISPR/Cas9, we were able to produce an, in principle, unlimited number of functional HES-2-derived cardiomyocytes with fluorescently labeled Z-lines. We generated two genomically modified HES-2-derived stem cell lines, namely HES-2-Citr-Linker and HES-2-Citr-4Mut, in which the Citrine variant of YFP was fused to the C-terminus of sarcomeric α-actinin. In HES-2-Citr-Linker-CM, Citrine and ACTN2 were joined via a short (6 AS) linker peptide, whereas in HES-2-Citr-4Mut-CM, Citrine and ACTN2 were fused directly. Both cell lines yielded cardiomyocytes with functional and morphologically intact sarcomeric networks, in which 100% of the Z-lines were labeled with fluorescent Citrine (Section 3.2.4). Next to the ACTN2 reporter model which was published during the write-up of this thesis by Ribeiro et al. (2020), this work presented the first stable, endogenous fluorescent tagging of a cardiac Z-line protein in human PSC-derived cardiomyocytes. These ACTN2 reporter models presented the first robust, endogenous labeling of any protein

within the contractile apparatus in hPSC-CM, aside from the work published by Sharma et al. (2018), which was conducted simultaneously to this study, and where the authors demonstrated endogenous fluorescent tagging of cardiac titin.

The bright and highly photostable fluorescence of Citrine allowed for high-frequency, high-resolution confocal imaging of the sarcomeric network in live single-cell cardiomyocytes at 18.2 fps. In combination with our custom-written analysis algorithm, we were able to track Z-lines at roughly 50 nm spatial and 55 ms temporal resolution[11]. Endogenous tagging further facilitated longitudinal studies, such that e.g. the effects of drug treatment on sarcomere contractility could be assessed using the same batch of cells. Moreover, dual fluorescent imaging of sarcomere shortening and cytosolic Ca^{2+}-flux was possible in our transgenic cell lines.

Endogenous fluorescent tagging yielded supreme advantages over transient labeling via cell transfection. In a second approach to the fluorescent labeling of Z-lines in hPSC-derived CM (data not shown), we transfected hiPSC-CM with a plasmid using transfection reagent Viafect® (Anson 2015; Kaneko et al. 2016; Wang X et al. 2016). In the DNA construct, the Citrine variant of YFP was fused to α-actinin 2, i.e. similarly to our endogenously expressed fusion protein. Even though we could optimize these cell transfections to acceptable transfection efficiencies and cell viability, transfection efficiency was highly irreproducible and transient labeling was not suitable for longitudinal studies. Moreover, the transfection process itself seemed to interfere with the CMs' contractility, and multiple concerns such as overexpression with resultant artifacts occurred. In contrast, endogenous labeling provided ready-to-use, 100% fluorescent CM, and we did not encounter limitations often observed in transfected cells, e.g. the overexpression of the fluorescent protein (Section 1.6). Notably, the labeling was stable up to at least two years, and we were able to culture monolayers of HES-2-ACTN2-Citr-CM for at least four weeks without signs of newly arising functional deficits (data not shown).

Both, in terms of morphology and (single-cell) functionality, we observed no substantial differences between HES-2-Citr-Linker- and HES-2-Citr-4Mut-derived CM (Section 3.3.2). Considering that we have only studied one batch of differentiation per each cell line, no evaluation of whether the linker peptide affected functionality could be stated. As several benefits of linker peptides in the engineering of fusion proteins have been described elsewhere (Chen et al. 2013), we chose to focus on HES-2-Citr-Linker for follow-up investigations.

PCR sequencing of both cell lines revealed a single nucleotide base pair insertion (T:A) in the untargeted allele directly at the Cas9 cleaving site (Figure 3-16). This indel resulted in a

[11] Note that the scanning frequency and thus temporal resolution can easily be increased by a factor of 4 via bi-directional confocal scanning (x 2) and by scanning without line average (x 2). It can be increased even further by zooming into smaller ROIs, such as single myofibrils (see discussions in Section 4.6). However, all data in this thesis were recorded using uni-directional confocal scanning with two times line average, and in all cases, the whole micropatterned cardiomyocytes were imaged.

shift of reading frame, which in turn lead to a mutated wild-type allele with an extra 44 amino acids, as compared to the endogenous HES-2 wild-type sequence. Such indels near the Cas9 cleaving site during CRISPR/Cas9-mediated genome editing are likely to occur due to non-homologous end joining and have been described in the literature (Ratz et al. 2015; Uemura et al. 2016; Wang H et al. 2016). Although we did not observe obvious phenotypes in our 2D-culture experiments, further studies are required to test for potential impairments of sarcomere/cell functionality. In this context, the high expression level of the targeted allele (80%) vs. the mutated wild-type allele (20%; Figure 3-14) suggested partial malfunction in the latter.

To identify potential protein folding perturbations and help elucidate whether potential functional impairments were more likely to occur due to the mutation in the untargeted allele, or due to the fusion to Citrine itself, we generated 3D protein models as predicted via the web-based Phyre2 platform (Figure 3-17). These models predicted a C-terminal alpha-helix extension for the untargeted alleles, where the folding of the remaining protein seemed to be unaltered, as compared to the wild-type model. In the case of the knock-in alleles, the fusion to Citrine, with and without linker, appeared to induce a bending of the entire C-terminal end of ACTN2. Qualitatively, the fusion to Citrine thus seemed to induce a greater alteration of the ACTN2 structure than the extra alpha-helix in the mutated wild-type allele. In both cases, our protein models predicted no interference within the actin binding domain. Considering that ACTN2 forms dimers in the sarcomeric network (Ribeiro et al. 2014), either mutation might have interfered with e.g. the dimerization or strength of dimers, and thus might have resulted in reduced stability or resistance to tension. Further, interference within the EF hands located near the C-terminus might have affected the function and strength of the sarcomeres/myocytes, as these motifs are involved in the binding of titin (Ribeiro et al. 2014). Moreover, unspecific expression of GFP has been demonstrated to impair contractile function and isometric force development in rat myotubes and cardiomyocytes (Nishimura et al. 2006; Agbulut et al. 2007), which Agbulut et al. (2007) have attributed to the binding of GFP to the actin-binding site of myosin. However, it remains unclear whether this competition is also relevant when the fluorescent protein (in our case Citrine YFP) is bound to a protein (in our case α-actinin) rather than being expressed unspecifically.

Ultimately, a new generation of indel-free knock-in lines or Cas9-guided repair of the mutation would help clarify whether the extra 44 amino acids resulting from the indel affected contractile function. As stated above, the indel most likely occurred due to non-homologous end joining (Ratz et al. 2015; Uemura et al. 2016). In this regard, two simple changes to the CRISPR/Cas9 protocol may help inhibit NHEJ: a) Increasing the concentration of donor DNA and thus increasing the likelihood for HDR, and b) utilizing Cas9 ribonucleoproteins instead of Cas9 plasmids to avoid an exhausting induction of double-strand breaks (Wang H et al. 2016).

Last, potential off-target effects, which display a common concern in CRISPR/Cas9-mediated genome editing (Hsu et al. 2014), may have interfered with cell functionality. In this context, it is worthwhile to note that none of the four cell clones with a homozygous Citrine knock-in grew to sufficient clone sizes at the PSC level (Section 3.2.1): As the 2-isoform of α-actinin is believed to only be expressed in sarcomeric muscle (Perán et al. 2010; Shen et al. 2016), and thus not in undifferentiated stem cells, insufficient growth of these four homozygous knock-in stem-cell clones may be a hint to the occurrence of off-target effects in our cell lines, as our target mutation should have had no effect at the stem-cell level. Here, more homozygous knock-in clones would have to be studied, and an entire genome sequencing would help to further elucidate this consideration, however, such sequencing was not within the scope of this thesis.

Despite these potential impairments of *absolute* functionality and whatever reasoning lies behind them, we believe that our knock-in cell lines are suitable to study *relative* effects as probed in this thesis, where we compared the contractility of HES-2-ACTN-Citr-CM on substrates of various stiffness (discussed in Section 4.3), and the contractility under the influence of pharmacological intervention (discussed in Section 4.4).

4.2 Micropatterning for Defined Experimental Conditions

We employed two major techniques of micropatterning to shape HES-2-ACTN2-Citr-CM to physiological aspect ratios of 7:1 and cell sizes ranging from 700 to 2000 μm^2. To decipher the impact of matrix elasticity on sarcomere dynamics, we aimed to seed cells on very stiff glass substrates (E ~ 50-100 GPa) on the one hand, and on 15 kPa elastomer substrates on the other hand, which we chose to mimic the physiological microenvironment in the heart (Berry et al. 2006; Engler et al. 2008). On relatively hard glass substrates, straightforward conventional PDMS stamping yielded well-defined and reproducible micropatterns of rectangular Synthemax® islands (Figure 3-4). On soft 15 kPa elastomer substrates, however, conventional PDMS stamping did not yield sufficient results, as immersion of the stamps into the elastomer resulted in distorted and irreproducible patterning geometries (Figure 6-3). We thus turned to a technique first described by Hampe et al. (2014), in which the ECM proteins were deposited through cavities in microstructured lift-off epoxy membranes (Hampe et al. 2014). Optimizing this lift-off membrane technique to our needs (Section 2.3.5) rendered micropatterning of hPSC-CM on 15 kPa elastomer substrates sufficient (Figure 3-5).

Micropatterning of HES-2-ACTN2-Citr-CM facilitated the convenient studying of sarcomere dynamics in single cells under defined experimental conditions. In single-cell experiments, a defined cardiomyocyte shape is crucial, as sarcomere alignment is known to be regulated by cell shape, and an optimized contractility of CM at physiological aspect ratios of 7:1 has been reported (Bray et al. 2008; Kuo et al. 2012; McCain et al. 2014; Ribeiro AJS et al. 2015). We examined micropatterned CM a) in contractility experiments on substrates

of different stiffness (see discussions in Section 4.3) and b) in contractility experiments testing for drug interference (Section 4.4). While micropatterning certainly was of great help in assessing the impact of matrix elasticity on sarcomere dynamics in isolated CM, it remains questionable whether investigating micropatterned single-cell CM is reasonable in pharmacological screening experiments. Single-cell cultures, and 2D cultures with apical-basal polarity in general, have repeatedly shown to be highly unphysiological, nowhere close mimicking the 3D *in vivo* microenvironment in the heart (Baker and Chen 2012; Kurokawa and George 2016; Atmanli and Domian 2017; Duval et al. 2017; Fang and Eglen 2017). To this end, turning to organized monolayers or more advanced 2D culture formats (Salick et al. 2014), 3D culture (Fong et al. 2016; Zuppinger 2016), or even engineered tissue/myocardium (Tiburcy et al. 2014; Kurokawa and George 2016; Tiburcy et al. 2017) might be more suitable in terms of pharmacological testing. In these culture formats as well, our novel cell lines with endogenous fluorescent labeling can potentially provide insight into sarcomere dynamics or intercellular synchronicity, and into how these parameters are influenced when exposed to medical drugs.

Keeping these considerations in mind, the results and interpretations discussed below may not transfer to larger ensembles of CM including monolayers, 3D cultures, tissue or even the heart in whole. Nonetheless micropatterning will be of use to upcoming follow-up studies, e.g. in contractility experiments with pairs of CM seeded at various spacing, where we aim to understand mechanisms of intercellular cross talk and synchronization (Section 2.3.1, Figure 2-4-B).

4.3 The Effects of Extracellular Matrix Elasticity on Cardiomyocyte Contractility

Aiming to understand the influence of extracellular matrix elasticity on the dynamics of sarcomeric beating, we ran contractility experiments with micropatterned HES-2-ACTN2-Citr-CM on stiff glass and relatively soft 15 kPa elastomer substrates. We custom developed a Python®-based analysis algorithm to extract a) basic parameters of sarcomeric beating and b) the inter-sarcomeric cross-correlation of contraction amplitudes as a measure for the coherence of sarcomeric beatings (for a detailed description of the algorithm, see Section 3.3.1). Combining measurements conducted with CM derived from both, HES-2-Citr-Linker and HES-2-Citr-4Mut, we analyzed 20 cells on glass (including a total of 216 sarcomeres and 268 beating periods, corresponding to a total of 2850 individual sarcomere contractions), and 16 cells on 15 kPa (177 sarcomeres, 213 beating periods, 2270 individual sarcomere contractions).

We observed significantly higher sarcomere contraction amplitudes on 15 kPa substrates, where the average contraction amplitude, ΔSL_{max}, was roughly double as high as compared to glass ($\Delta SL_{max,\,glass}$ = 0.09 ± 0.03 µm/0.10 ± 0.01 µm [Linker/4Mut], $\Delta SL_{max,\,15\,kPa}$ = 0.20 ± 0.02 µm/0.19 ± 0.06 µm, $p < 0.001$, Table 3-3 and Table 3-4). Similarly, the maxi-

mal observed contraction amplitudes, Max. ΔSL_{max}, were higher by 57-95% on 15 kPa (Max. $\Delta SL_{max, glass}$ = 0.20 ± 0.04 µm/0.23 ± 0.07 µm, Max. $\Delta SL_{max, 15\ kPa}$ = 0.39 ± 0.04/ 0.36 ± 0.07 µm, $p < 0.001$). Similar findings of reduced cardiac contraction amplitudes on stiff extracellular matrices have been described in the literature. Engler et al. (2008) examined embryonic quail and chicken CM on PA gels of various stiffnesses, and found a dramatic loss in contractile function when the substrate's Young's modulus E exceeded that of the normal heart by far. Notably, Engler et al. (2008) did not analyze contractions on the sarcomere level, but evaluated global contractions of the cells in bright field microscopy and mapped these to the motion of fluorescent particles embedded in the gels. More similar to our study, Hersch et al. (2013) transfected late rat embryonic cardiomyocytes with an ACTN2-GFP construct and analyzed sarcomere contractions on PA gels of stiffnesses 15-90 kPa. Interestingly, despite there being a trend towards smaller contraction amplitudes with increased ECM stiffness, Hersch et al. did not report a significant difference between 15 kPa (4.7 ± 2.1%), 30 kPa (4.0 ± 2.9%) and 90 kPa (3.0 ± 1.7%; Hersch et al. 2013). However, sarcomere shortenings of roughly 4% seemed low as compared to other studies with rat cardiomyocytes, where contraction amplitudes were ~ 0.15 to 0.20 µm corresponding to roughly 10% shortening (Galie et al. 2013; Shintani et al. 2014), and thus similar to our observations (~ 10% average shortening on 15 kPa, ~ 6% on glass). A reasonable explanation for these findings would be that CM on very stiff glass substrates are forced to perform contractions in a somewhat isometric regime, whereas on softer 15 kPa elastomers, contractions are more isotonic.

For both cell lines, we observed a slight tendency towards higher beating frequencies on 15 kPa, however, the difference was not significant. These observations were in line with those of Engler et al. (2008), who observed a decay in beating frequency for CM seeded on stiff matrices. To further test whether matrix stiffness also affected consistency of the beating period, we evaluated the cell-specific relative standard deviation of the beating period, Rel. $STD_i(T)$, i.e. the fluctuation of the period of spontaneous beatings within the individual cells. Here, no significant difference between glass and 15 kPa could be found for either one of the cell lines.

No difference between glass and 15 kPa could be observed in the time-to-peak contraction, T_{max}. Interestingly, the fluctuation of T_{max} within the individual cells, Rel. $STD_i(T_{max})$, was significantly lower on 15 kPa in the case of both cell lines. These observations go along with a higher level of coherence of sarcomeric beating on 15 kPa (see below), and a possible explanation could involve that on stiff glass substrates, mechanisms of pulling apart between competing sarcomeres yielded some sarcomeres, which quickly reached their shortest lengths, whereas other sarcomeres performed a full, lengthy contraction during the same beating period.

Next, we wanted to gain insight into how matrix elasticity affected the level of coherence of sarcomere contractions. To this end, we turned to analyzing the cross-correlation of the

change in sarcomere length, ΔSL, between the individual sarcomeres within single myofibrils (see Section 3.3.1 for details). By calculating the correlation coefficients at the various inter-sarcomeric distances, we obtained a measure for the coherence of contractions at the subcellular level. We observed highly coherent contractions on 15 kPa substrates at inter-sarcomeric distances up to at least twelve sarcomeres[12]. On glass, we typically observed positive correlation only between next-neighbor sarcomeres and correlation quickly dropped to almost zero correlation even for next neighbors (Figure 3-26), and, notably, even anti-correlation occurred (Figure 3-25-A & -C). Observation of anti-correlation on these stiff glass substrates coincided with the qualitative observation of pulling-apart mechanisms between several sarcomeres, and all together, our results suggested that the rigid mechanical microenvironment of the myocytes put constraints on the contractions of the intracellular contractile elements and forced those sarcomeres into competition. This "pulling-apart" between sarcomeres is illustrated qualitatively in the next-neighbor correlation diagram in Figure 4-1-A, which shows the stereotypical contraction patterns (Section 3.3.1) of four neighboring sarcomeres in a representative HES-2-Citr-Linker-CM on glass. While sarcomere #1 and #2 were contracting, sarcomere #3 got elongated, which resulted in anti-correlation between the two neighboring sarcomeres #2 and #3. In contrast to that, Figure 4-1-B illustrates homogenous beating of a representative HES-2-Citr-Linker-CM on 15 kPa, where the stereotypical contraction patterns indicate correlated contractions of all four neighboring sarcomeres.

As this study, to our knowledge, presented first similar insight into sarcomere dynamics, comparing our findings to previous statements in the literature seems impracticable, particularly when considering that (inter-)sarcomeric coherence appears to not have been previously assessed. However, linking higher substrate stiffness to the incoherence of sarcomeric beating intriguingly raises the question of whether higher ECM rigidity per se might play a role in the pathogenesis of heart failure and/or cardiac arrhythmia. One pathophysiological implication of the observed incoherencies on stiff substrates might be that these directly affect the electro-mechanical coupling in the heart. Various mechanisms of decreased mechanical output and ultimately heart failure resulting from fibrosis have been well-recognized, both on the cellular and the global scale (Engler et al. 2008; Ribeiro AJS et al. 2015; Boothe et al. 2016; Gabriel-Costa 2018). To our knowledge, however, no mechanism of (inter-)sarcomeric incoherencies has been noted to play a role in the impairment of the electro-mechanical coupling in a stiff fibrotic ECM environment. In this regard, our cell lines and data might provide a first (dynamic) subcellular insight into this aspect of why work generation is less efficient in the fibrotic heart.

[12] Data in this thesis are displayed for only up to 7-sarcomere distances (i.e. eight sarcomeres), as no longer myofilaments could be analyzed in some of the individual cells.

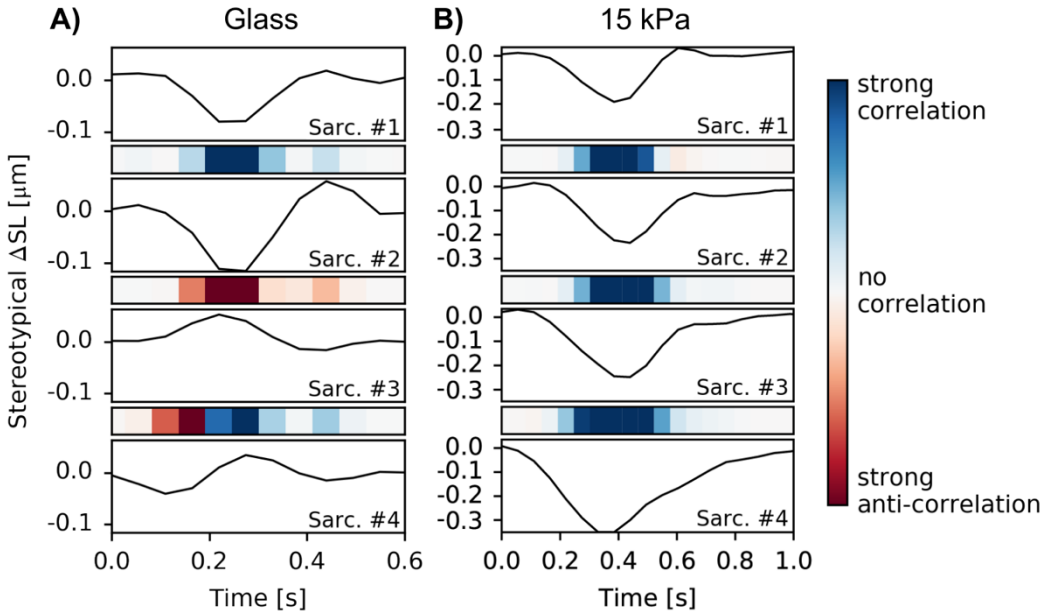

Figure 4-1: Competition between neighboring sarcomeres on stiff glass substrates. The next-neighbor correlation diagrams show the stereotypical contraction patterns (see Section 3.3.1 for definition) of four neighboring sarcomeres in a representative HES-2-Citr-Linker-CM on glass **(A)** and on 15 kPa **(B)**.

To further elucidate whether the above-described effects were only perceivable when comparing the extremes of physiologically stiff elastomeres (15 kPa) to very stiff, unphysiological glass substrates (50-100 GPa), our group conducted a follow-up study, where the experimental design was extended to elastomer substrates with four different Young's moduli ranging from 7 to 60 kPa, and thus including that of the infarcted heart (Berry et al. 2006). In this follow-up study, mainly performed by physics PhD candidate Daniel Haertter, M.Sc., the above findings could indeed be confirmed in this (patho-)physiological range of stiffness (manuscript in preparation).

Importantly, this thesis lacks a morphological characterization of the sarcomeric organization within HES-2-ACTN2-Citr-CM. Safran and colleagues have discussed that the correlated beating in cardiomyocytes might be limited by the structural order of myofibrils (Majkut et al. 2013; Dasbiswas et al. 2015). Considering this, it may very well be possible that the observed differences between 15 kPa and glass substrates can at least partially be attributed to different ordering of the contractile elements due to the altered stiffness of the microenvironment. However, in a brief qualitative cross-check we did not observe any striking differences in myofilament density or axial alignment, and we thus believe that structural constraints were at least not entirely accountable for inducing the observed functional effects. This would be supported by Hazeltine et al. (2012), who observed similar effects of substrate stiffness on CM contractility, and could not link these effects to morphology. On the other hand, we did observe what appeared to be *bridged* sarcomeres in

many of the micropatterned cells. In these bridged sarcomeres, longitudinal actinin-*bridges* seemed to be spanning between two neighboring Z-lines, and these bridges were oriented perpendicularly to the regular Z-line orientation, i.e. in parallel with the myofilament (for example, see Figure 3-11-A). While we have observed at least a few of these bridged sarcomeres in the axial periphery of almost all micropatterned CM, it seems like the frequency of appearance of bridged sarcomeres in the center-part of the cells was higher on glass substrates. Interestingly, we barely observed bridged sarcomeres in confluent monolayers or larger clusters of HES-2-ACTN2-Citr-derived CM (Figure 3-9). This purely qualitative observation may indicate that bridged sarcomeres somehow arose from boundary conditions imposed by the unphysiological micropatterning environment, and the lower rate of bridged sarcomeres on 15 kPa would be an indirect sign that these constraints were more severe on glass. From these considerations, two helpful characterization analyses can be proposed: a) Defining an order parameter for Z-lines similar to that described by Zemel et al. (2010) for acto-myosin filaments seems to be an appropriate measure for the overall organization and alignment of myofilaments. This analysis should also derive the amount/fraction of bridged sarcomeres. b) A potential correlation between the contraction amplitude of bridged and non-bridged sarcomeres should be examined.

As the main focus of this thesis was in the development of the fluorescence indicator system and the respective analysis algorithm per se, and as it was thus simply not possible to recruit more data within in the time scope of this thesis, the data described here cannot be utilized to draw final conclusions on the discussed effects. However, even these preliminary data certainly do underline the crucial need for optimal culture conditions of hPSC-derived cardiomyocytes, which has repeatedly been stressed in the literature to ensure optimal maturation and functioning of the myocytes (Parker et al. 2008; Zhang et al. 2013; McCain et al. 2014; Tiburcy and Zimmermann 2014; Ribeiro AJS et al. 2015).

4.4 Drug Testing Applicability Exemplified Using Omecamtiv Mecarbil

One potential and intriguing use of our novel indicator system would be an application in high-throughput drug screening. To show proof of concept of such drug testing applicability, we ran longitudinal contractility experiments in the presence of the selective cardiac myosin activator omecamtiv mecarbil (OM) at difference concentrations. On 15 kPa, OM concentrations ranged from 10 nM to 10 μM, thus including previously described EC_{50} values (10-100 nM; Malik et al. 2011; Woody et al. 2018) as well as clinically relevant plasma concentrations (100-600 nM; Vu et al. 2015). At micromolar [OM], sarcomeric beating became highly inhomogeneous, and at 10 μM most of the sarcomeres entered into what appeared to be a state of constant (hyper-) contracture. At these micromolar concentrations, our Python®-based analysis algorithm did not apply to determine the starting time points of contraction, $t_{start,i}$, as well as the time points of peak contraction, $t_{max,i}$, and thus the

individual contraction amplitudes $\Delta SL_{max,i}$ with their current definition (see Figure 3-19, Section 3.3.1 and Section 4.7). Therefore, only concentrations up to 100 nM OM were included in the analysis of ΔSL_{max} and related parameters. Notably, the analysis algorithm did reliably return the sarcomere length SL(t) and the change in sarcomere length ΔSL(t) over time in the full concentration range.

In the nanomolar range, we qualitatively observed an expected positive inotropic effect of OM on sarcomere contractions (see kymographs in Figure 3-28). This notion was supported quantitatively by an observed decrease in the minimal sarcomere length (Min. SL) with increasing [OM] as a sign of hypercontraction, and this decrease in Min. SL continued for micromolar [OM]. At 100 nM, Min. SL was shorter by 4.3± 2.9%, and at 10 μM OM, it was shorter by 14.9 ± 4.3%, as compared to 0 nM OM ($p < 0.001$). Despite the observed decrease in Min. SL, no increase in sarcomere shortening amplitude could be registered directly by evaluating the contraction amplitude ΔSL_{max}, which did not change significantly in the nanomolar concentration range. We believe that this lack of an observed increase in ΔSL_{max} was most likely due to limited temporal resolution (18 fps) paired with limited statistics due to the relatively low number of analyzed cells. These limitations may have simply hindered the detection of the presumably small OM-evoked changes in sarcomere shortening amplitude. Here, an enhanced temporal resolution by a factor of 4 (i.e. ~ 70 fps), which is easily achievable with our reporter model and our existing setup (Section 4.6), would have rendered Z-line trajectories smoother and thus the evaluation of sarcomere contraction (amplitudes) more precise. With the current data, however, the effect of OM on sarcomere shortening amplitude could not finally be evaluated.

Next to the expected decrease in minimum sarcomere lengths during peak contractions, we observed a significant decrease in sarcomere rest lengths (SL_0) with increasing [OM]. These observations suggested that OM increased the passive tension of sarcomeres and/or decreased relaxation amplitude, which could ultimately be a sign of a negative lusitropic effect. An effect of OM on sarcomere rest length has not been described in the literature, but may be explained through its proposed mechanism of action, which includes prolonged actomyosin attachment and thin filament activation (Woody et al. 2018).

Liu et al. (2015) have further predicted slower – yet stronger – contractions of cardiac muscle in the presence of OM, based on an observed decrease in shortening velocity in *in-vitro* motility assays. In contrast, our evaluation of the time-to-peak contraction (T_{max}) yielded an insignificant trend towards shorter T_{max} in the nanomolar OM concentration range, while our analysis algorithm was not applicable to determine T_{max} in the micromolar concentration range (see above). Our observations supported those by Nagy et al. (2015), who have found a tendency towards faster contractions of permeablized rat CM up to the hundreds of nanomolar OM range despite increasingly slower global contractions at micromolar [OM]. To shine further light on this effect of OM with our indicator system a higher number of experiments at enhanced temporal resolution would be required.

An entirely unexpected observation was that of significantly higher contraction frequencies with increasing [OM] up to 150% baseline for micromolar concentrations ($p < 0.001$). To our knowledge, no OM-mediated increase in contraction frequency has been described in the literature, nor do the described mechanisms of action of OM predict a direct effect on beating frequency. Concentration-dependent increases in heart rate have been described for other downstream calcium sensitizers such as Levosimendan, but these most likely occur due to effects on upstream regulatory mechanisms (Bokník et al. 1997; Toller and Stranz 2006). The observed increase in beating frequency in our single-cell experiments thus remains unclear and may not be directly OM-evoked, but may rather be attributed to the single-cell cardiomyocytes adapting to some chemical alterations imposed by the titration of OM into the culture media. Here, control re-runs of these contractility experiments in the presence of OM with a different batch of differentiation might certainly help to clarify this potential effect. But notably, a significant increase was not only observed on 15 kPa, but also in a second set of experiments on glass and already at relatively little OM concentrations of 1 nM (Figure 6-8). In this regard, it seems tempting to test whether CM monolayers or even engineered tissue would respond similarly to OM exposure.

We next turned to assessing the effect of OM on inter-sarcomeric coherence of contractions. In a qualitative video analysis, we observed a clear trend towards more inhomogeneous sarcomere contractions with increasing [OM], which became particularly evident at the step from 100 nM to 1 μM OM (compare kymographs in Figure 3-28). Our quantitative correlation analysis corroborated this notion (Figure 3-32), however, a higher number of experiments would again be required to evaluate whether this effect was relevant at nanomolar concentrations. The drop to almost zero correlation at 1 μM OM was in line with our qualitative observations of highly inhomogeneous contractions, and the drop to negative values, i.e. anti-correlation, at 10 μM OM reflected the observation that in this *state of constant contracture*, the sparsely occurring contractions of some few sarcomeres within a myofilament would result in a pulling-apart/rupture of the neighboring sarcomere(s). To our knowledge, no such interference of OM with the coherence of contractions has been described in the literature. Whatever reasoning lies behind the OM-mediated incoherencies, they might certainly affect the global homogeneity of contractions within the heart. However, since no such observations have been described, it might again be possible that these effects do not transfer to greater ensembles of cardiomyocytes.

It is noteworthy to further discuss what happened at micromolar [OM], as *i)* clinically relevant plasma concentrations come close to the low micromolar range (100-600 nM; Vu et al. 2015), and *ii)* in multiple previous publications the effects of OM were studied up to these micromolar concentrations (Malik et al. 2011; Nagy et al. 2015; Swenson et al. 2017; Woody et al. 2018). As already briefly stated above, at 1 μM OM sarcomere contractions became dramatically more inhomogeneous as compared to 100 nM OM, and it qualitatively seemed like neighboring sarcomeres competed in pulling each other apart. This resulted in Z-line movements that resembled a wiggling around rather than coherent contractions

along the myofilaments, despite still globally contracting cardiomyocytes. At 10 μM, sarcomere contractions barely occurred at all, and most of the sarcomeres appeared to be in a state of constant (hyper-) contracture with sarcomere lengths as short as 1.1 μm. Some few sarcomeres, however, were found to be in a largely stretched state (up to 3.8 μm), and occasionally, we would observe contractions of single sarcomeres into the above-mentioned state of constant contracture, which resulted in other, neighboring sarcomeres being dragged apart. The most reasonable explanation of such observations would be, that at micromolar [OM], the OM-enhanced entry into the strongly bound state of the contractile cycle and the OM-enhanced prolongation of this state were (sub-) saturated, which resulted in a duration of contraction that exceeded a single full contraction cycle (Woody et al. 2018). In this regard, our observations at micromolar [OM] could explain a decrease in cardiac force development at OM concentrations beyond 1 μM, as observed by Nagy et al. (2015). Furthermore, Woody et al. (2018) demonstrated an almost entirely diminished myosin working stroke at 10 μM OM albeit a high level of thin-filament activation. Here, these findings may be in line with our observation of a state of constant contracture at 10 μM OM.

Notably, the above interpretations certainly remain vague. Next to a higher number of experiments for improved statistics at enhanced temporal resolution, there would be a need for several biochemical and electrophysiological co-experiments to decipher the complex effects of OM on sarcomere dynamics. However, even these rather rudimental experiments demonstrate the applicability of our indicator system to drug testing *in vitro*. With regard to omecamtiv mecarbil, it seems like our indicator model contribute an important morphological feedback in live CM to the research community.

4.5 Studies of Myofibrillogenesis in Live Cardiomyocytes

Having cell lines with an endogenous label of a sarcomeric network protein at hand raised the intriguing prospects to visualize sarcomere development and myofibrillogenesis in live CM. In brief proof-of-concept experiments, we tested the applicability of our system to image a) the assembly and b) the disassembly of cardiac myofibrils.

To image the development of the sarcomeric network, we plated HES-2-ACTN2-Citr-CM on unpatterned Synthemax®-coated substrates after Accutase®-digest and recorded time-lapse z-stacks of isolated single-cells (Figure 3-33). As previously described by Dabiri et al. (1997), sarcomere assembly would start out with Z-bodies that were gradually drawn towards the growing myofilaments and integrated into new sarcomeric structures. Typically, the first contracting sarcomeres became visible within one to two hours, however, in some cells we observed well-organized, contracting myofibrils as early as 30 minutes after seeding. We refrained from further investigating the details of sarcomere assembly within the scope of this thesis but noted the potential insight that our knock-in cell lines can contribute to the understanding of myofibrillogenesis.

It was further interesting to image the disassembly of sarcomeres during the digestion process, as digestion of CM is a common part of the experimental procedure during the preparation of experiments as well as during cell culture. To this end we recorded time-lapse confocal z-stacks of HES-2-ACTN2-Citr-CM during Accutase® digest and found that the cardiac myofibrils appeared to be axially coiled up during the detachment process, while the sarcomeres seemed to stay morphologically intact (Figure 3-34). We did not further track the fully dissociated cells in suspension, but if the sarcomeres really did stay intact, it would yield important insight to examine whether these *digested* sarcomeres/myofibrils were still able to contract.

4.6 Improvement of Temporal Resolution

One major drawback of this thesis was that the Python®-based analysis algorithm was not available until one year after the above presented data were recorded. At the time when we video recorded the presented contractility experiments, we a) expected to analyze the entire cells in whole rather than just single myofilaments, and b) we adjusted our microscope settings such that it would yield a fairly good compromise between contrast and temporal resolution (18.2 fps, which seemed sufficient for gross qualitative observations by eye). Having employed our analysis algorithm to our data, we now know that the contrast obtained with the bright fluorophore Citrine sufficed by far, however, a temporal resolution of 18.2 fps was barely enough to precisely determine sarcomere shortening and related parameters, particularly when seeking to detect subtle differences e.g. during drug treatment. In retrospect, we would have thus shifted our focus away from improving contrast and more towards temporal resolution. We would have refrained from scanning with two times line average (2-fold increase of temporal resolution), and we would have employed bi-directional scanning, which yields an extra factor of two in temporal resolution. In total, the temporal resolution can thus easily be increased 4-fold to roughly 72 fps with our existing system, which would greatly enhance the precision of parameters such as contraction amplitude, time-to-peak contraction (T_{max}), and shortening/relaxation velocity. Further, for analysis of sarcomere contractility as performed in this thesis, it would be sufficient and thus reasonable to zoom into the single myofilaments of interest. This would not only reduce the number of line scans and thus greatly improve temporal resolution even further, but it would also increase pixel- and thus spatial resolution, if needed. An additional recording of the entire cell in whole should, however, still be pursued for potential further analysis of the global contraction behavior.

4.7 Shortcomings of and Potential Improvements to the Analysis Algorithm

Although our Python®-based analysis algorithm was powerful in the tracking of Z-lines and in deducing several relevant parameters of sarcomere contractility, it does leave room for several improvements.

In general, the semi-manual manner of the algorithm rendered the analysis yet very time consuming (30-45 minutes per each cell). The major manual steps included *i)* selection of the myofilament of interest in ImageJ-based Fiji, *ii)* adjustment of thresholds and several other parameters for the intensity-peak-finding algorithm for the tracking of Z-line trajectories, and *iii)* manual picking of the starting time point of each contraction cycle, respectively. Such time-consuming analysis may be feasible when analyzing a limited number of cells, as in this thesis, but particularly when considering high-throughput screening experiments, a fully automatized algorithm would be needed. In terms of the above-mentioned manual steps, the following adjustments could be implemented for automation: Regarding *ii)*, a better temporal resolution would allow for a different type of peak finding algorithm, where adjustments of parameters would only have to be set once for an entire set of experiments: For relatively "low" temporal resolution, as for the data presented in this thesis (18.2 fps), the assignment of intensity peaks to Z-lines for many cases could only be achieved via careful tuning of settings such as minimum peak distance or minimum peak height. At higher temporal resolution, the assignment of intensity peaks to Z-lines can instead be accomplished via a differential method, where the peaks are automatically assigned based on relative, incremental movement. Regarding *iii)*, the starting point of each contraction cycle could be derived from frame-to-frame auto-correlation, i.e. cross-correlation of frame i and i+1. A drop of this auto-correlation below a certain threshold would then indicate a relevant motion and could be utilized to define time frames of starting contractions. Regarding *i)*, the extent of potential automation depends on the application. For detailed studies of sarcomere contractility on the sub- to intercellular level, a manual selection of filaments seems appropriate. To speed up the analysis, a graphical user interface could here be implemented directly into the Python® script, such that there would be no need for working with ImageJ/Fiji. If, on the other hand, a larger scale analysis is of interest, e.g. in high-throughput screening, the tracking algorithm may have to be changed entirely.

A further deficit of our analysis algorithm is that it does not reliably deduce parameters such as contraction amplitude or time-to-peak contraction for very incoherent beatings. If the starting time points of each contraction cycle j largely differ among sarcomeres i, $t_{start,j}$ and possibly even $t_{max,j}$ will not be accurate for all sarcomeres i and contraction cycles j (see Figure 3-19). At least the determining of the contraction amplitudes ΔSL_{max} would then be more accurate if determined as $\Delta SL_{max,i,j} = SL_{0,i} - SL_{min,i,j} = SL_{0,i} - SL_i(t_{max,j})$, where $SL_{0,i}$ is the average rest length of sarcomere i during the recording time, and $SL_{min,i,j}$ is the minimum sarcomere length of sarcomere i during contraction cycle j. Currently ΔSL_{max} for sar-

comere i during contraction cycle j is calculated as $\Delta SL_{max,i,j} = |\Delta SL_i(t_{max,j}) - \Delta SL_i(t_{start,j})| = SL_i(t_{start,j}) - SL_i(t_{max,j})$, where the determining of $t_{start,j}$ and thus $SL_i(t_{start,j})$ for each contraction cycle j adds an extra uncertainty (see Figure 3-19), as one would expect $SL_{0,i}$ to be constant over time.

Several further deficits of our analysis were, again, rather due to the "low" temporal resolution. At higher temporal resolution, the following improvements could be implemented for further insight: a) evaluation of contraction and relaxation velocities, b) analysis of dwell time in state of full contraction, and c) correlation analysis of the first derivative of sarcomere shortening, where the latter should be more powerful in detecting subtle differences in the coherence of beatings than correlation analysis of just the contraction amplitudes. As stated in Section 4.6, the temporal resolution can easily be increased 4-fold with our existing imaging setup by simple adjustments to standard recording settings, and by that, the full power of our video analysis algorithm could be utilized.

4.8 Concluding Remarks and Perspectives

In conclusion, we have demonstrated stable, endogenous and heterozygous fluorescent labeling of sarcomeric actinin in HES-2-derived cardiomyocytes using CRISPR/Cas9. Endogenous labeling allowed for the generation of an, in principle, unlimited number of live and functional CM with fluorescent Z-lines. Second, we have established two well-described micropatterning techniques in our laboratories and optimized these techniques to our needs, such that we were able to efficiently micropattern CM on very stiff glass and on softer elastomer substrates – in our case 15 kPa to mimic the physiological microenvironment of the heart. Third, we have developed a powerful Python®-based video analysis algorithm to derive several relevant parameters of sarcomere contractility including a measure for the inter-sarcomeric coherence of contractions. Taken together, our indicator system allowed for exciting insight into sarcomere dynamics at the subcellular level, and particularly due to the robust endogenous labeling, we were able to examine these dynamics in longitudinal studies, which open prospects for high-throughput analysis. Despite several limitations to the genomically modified cell lines that may render *absolute* values unreliable, we believe that our knock-in cell lines do allow for the evaluation of *relative* effects.

Our experiments yielded insight into how extracellular matrix elasticity affects sarcomere contractility and coherence of contractions. Not only were we able to validate a well-known impairment of contractile function on very stiff glass substrates, which became most strikingly evident in a decrease of contraction amplitudes by roughly 50% as compared to 15 kPa, but we also demonstrated a severe impact of ECM elasticity on inter-sarcomeric coherence. We here observed highly coherent contractions on 15 kPa substrates at inter-sarcomeric distances up to at least twelve sarcomeres, whereas on stiff glass substrates, correlation quickly dropped to almost zero correlation even for next neighbors, and, notably, even anti-correlation occurred. Observation of anti-correlation on these stiff glass sub-

strates coincided with the observation of pulling-apart mechanisms between several sarcomeres, and all together, our results suggested that the rigid mechanical microenvironment of the myocytes put constraints on the contractions of the intracellular sarcomeres, that forced these contractile elements into competition.

We demonstrated proof-of-concept applicability of our indicator system to drug screening experiments by running contractility experiments in the presence of the cardiac myosin activator omecamtiv mecarbil. Our observations supported a positive inotropic effect of OM, and interestingly, our data further suggested that OM a) has an undescribed positive chronotropic effect, and b) impairs inter-sarcomeric coherence of contractions in isolated, single-cell CM. Mainly due to a limited number of analyzed cells and thus poor statistics, and due to an insufficient temporal resolution, a more sophisticated interpretation of these proof-of-principle experiments was hindered.

In simple time lapse experiments of live CM, we moreover demonstrated proof of concept of how our indicator system can possibly contribute to the understanding of the assembly and disassembly of cardiac myofibrils.

We have discussed multiple shortcomings of this study and suggested potential improvements accordingly. Perhaps most strikingly, the temporal resolution of roughly 18 fps appeared to be insufficient particularly when seeking to evaluate subtle effects of pharmacotherapy. Importantly, this temporal resolution can easily be increased 4-fold with our existing imaging system, which should render our indicator system sufficiently sensitive to detect small changes in sarcomere contractility.

Our proof-of-principle experiments raised intriguing prospects of multiple follow-up studies. First, the evaluation of the effects of ECM elasticity should be extended to shine further light on how the boundary conditions imposed by the cells' microenvironment affect sarcomere contractility, including the mechanical interplay between sarcomeres. As stated in Section 4.3, our group has, in fact, already conducted such follow-up study, where the experimental design was extended to elastomer substrates with four different Young's moduli ranging from 7 to 60 kPa. In this study, mainly performed by PhD candidate Daniel Haertter, M.Sc., we were able to validate the above-discussed effects for this physiologically more relevant elasticity range, and we further proposed an elastic spring model, that could explain the observed incoherencies and even anti-correlation on stiffer substrates (manuscript in preparation). Notably, these findings are not only relevant to the cardiology field, but also provide insight into the biophysics of basic cell mechanics.

Second, our indicator model may yield important insight in pre-clinical pharmaceutical screening experiments, while pre-clinical screening experiments with hPSC-CM offer the advantage of avoiding potential side effects. Here, our indicator system could potentially be used in high-throughput screening, if the analysis can be further automatized. Third, the endogenous ACTN2 fluorescence reporter may provide exciting insight in studies on the development of the sarcomeric network both, in matured CM and during the maturation of

cardiac progenitors. More elaborate time lapse imaging of live, matured HES-2-ACTN2-Citr-CM may yield further insight into myofibrillogenesis, and time lapse imaging during the maturation of HES-2-ACTN2-Citr-derived cardiac progenitors can provide structural insight into the maturation process. Fourth, micropatterning of HES-2-ACTN2-Citr-CM offers the potential to study length- and time scales of synchronization between cardiomyocytes. To this end, our photomasks already include patterns, which allow us to seed cardiomyocytes pairwise at various distances and geometries (Figure 2-4-B). Further, optical trapping experiments with HES-2-ACTN2-Citr-CM should yield exciting mechanical and morphological insight into the time scales of the development of sarcomeres and/or myofibrils under stretch. As mechanical stress has been identified as one of the major elicitors and modulators of myofibril formation (Russell et al. 2010; Yang et al. 2016; Yuan et al. 2017), it is highly intriguing to test the real-time response of our knock-in CM to mechanical triggers. One way to apply mechanical stress and record direct mechanical feedback is via optical tweezers. Our existing resonant-scanning confocal microscope setup combines optical trapping and high-speed fluorescence imaging (Schlosser 2015). In particular, we here hope to gain insight into the cells' real-time response to stress, and we seek to explore whether – and if so, on which time scales – myofibrillogenesis can be triggered by a pulling via the optical tweezers. Last, it is intriguing to generate further knock-in cell lines from hiPSC of patients with defined mutations. We have hiPSC cell lines available from patients e.g. with mutations in either the alpha-cardiac actin gene or the gene encoding for ryanodine receptor 2. Employing CRISPR/Cas9, it should be relatively straightforward to knock in the Citrine gene into the ACTN2 locus of these hiPSC lines. Comparing contractility assays with CM derived from these mutant lines with those of CM derived from our "wild-type" knock-ins may provide revealing insight into how those mutations of the sarcomeric apparatus or the Ca^{2+}-storage system affect sarcomere contractility, which may add a whole new aspect to cardiovascular disease modeling.

5 Summary

The aim of this thesis was to develop a fluorescence indicator system for the *in-vitro* characterization of human stem cell-derived cardiomyocytes. We hypothesized that a) the Z-lines in hPSC-CM can be reliably labeled via an endogenous expression of an ACTN2-Citrine-fusion protein, b) endogenous labeling of Z-lines allows for longitudinal studies on sarcomere contractility and sarcomere development, c) fluorescently labeled CM can be plated for high-throughput analysis under defined experimental conditions via micropatterning on substrates of defined elasticity, and d) our indicator system can be applied in drug testing.

Employing the CRISPR/Cas9 system, we generated two human embryonic stem cell lines with a heterozygous knock-in of Citrine into the sarcomeric actinin gene locus. In one of the two lines, Citrine was fused directly to the C-terminal end of ACTN2, whereas we introduced a small linker peptide in the second line. Exploiting our group's knowledge in cardiac differentiation, we were able to derive an, in principal, unlimited number of functional cardiomyocytes with a stable and bright fluorescent label of Z-lines from both cell lines. We turned to micropatterning and shaped the fluorescing myocytes to defined single-cell geometries on both, very stiff glass substrates, and 15 kPa elastomers. In so-called *contractility assays*, we then video-recorded the beating micropatterned CM at high spatial and temporal resolution using our high-speed resonant-scanning confocal microscope. Finally, we employed a custom-developed Python®-based video analysis algorithm, which enabled us to derive relevant parameters of sarcomere contractility including inter-sarcomeric coherence of contractions.

The combination of the above techniques provided unprecedented insight into sarcomere dynamics at the subcellular level. We demonstrated a severe impairment of contractile function on very stiff glass substrates, which resulted in a 50% decrease in sarcomere shortening amplitude, as compared to 15 kPa. Further we observed a strikingly lower level of inter-sarcomeric coherence of contractions on these stiff substrates, which we attributed to the boundary conditions imposed by the microenvironment of the cells. Next, we demonstrated proof of concept that our indicator system can contribute to drug screening assays by evaluating the effects of the cardiac myosin activator omecamtiv mecarbil on CM contractility. Last, we showed proof-of-principle applicability of our indicator system in the studying of myofibrillogenesis.

In conclusion, this study presents the first stable and robust, endogenous fluorescent labeling of a Z-disc protein in live hPSC-derived cardiomyocytes, next to the ACTN2 reporter model which was published during the write-up of this thesis by Ribeiro et al. (2020). In combination with high-speed confocal microscopy and our custom-developed, powerful video analysis algorithm, these knock-in CM provided exciting insight into sarcomere contractility and raised the intriguing prospects of multiple follow-up studies, including longi-

tudinal drug screening assays as well as studies that aim for a further understanding of basic mechanics of cardiac contractions.

6 Appendix

6.1 Additional Data and Data Illustrations

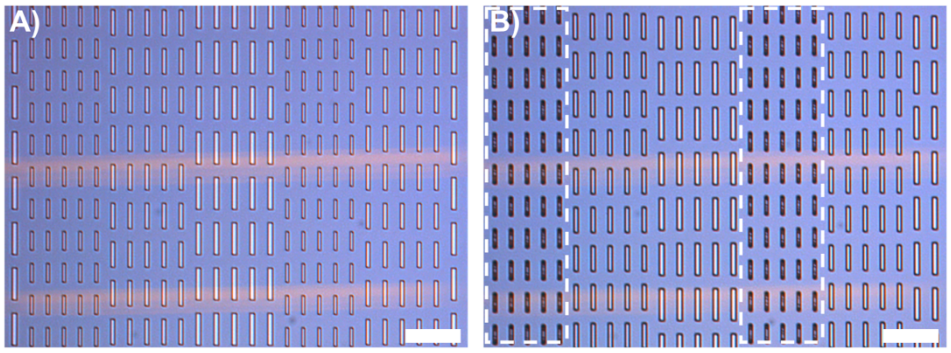

Figure 6-1: Optimization of the exposure time of 8 µm deep photoresist masters. (A) Exposure time: 3 s. **(B)** Exposure time: 8 s. The small structures on wafer B), as indicated by the white dashed lines, show signs of overexposure. Here, the photoresist could not fully be washed out during the photo development process. Images recorded with an upright bright-field microscope (Leica DLM 4000, Leica microsystems, Wetzlar, Germany). Scale bars 200 µm.

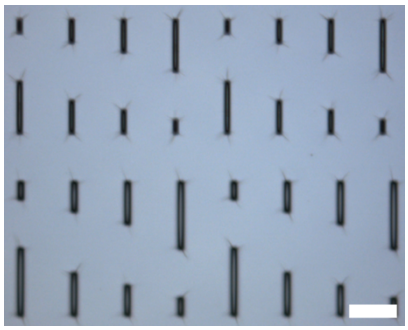

Figure 6-2: Incompletely developed 30 μm deep photoresist master. Wafer had been shaked in photo developer for 10 minutes. As the depth of the cavities (30 μm) was relatively large with respect to the width of the smallest rectangles (10 μm), simple shaking of the wafers in photo developer did not yield sufficient results. We here had to introduce an extra step of sonicating the wafers in photo developer for five minutes. Note that the pattern shown here was used during the earlier stages of this project, when we had not yet decided on the final shapes of the cardiomyocytes. Images recorded with an upright bright-field microscope (Leica DLM 4000, Leica microsystems, Wetzlar, Germany). Scale bar 100 μm.

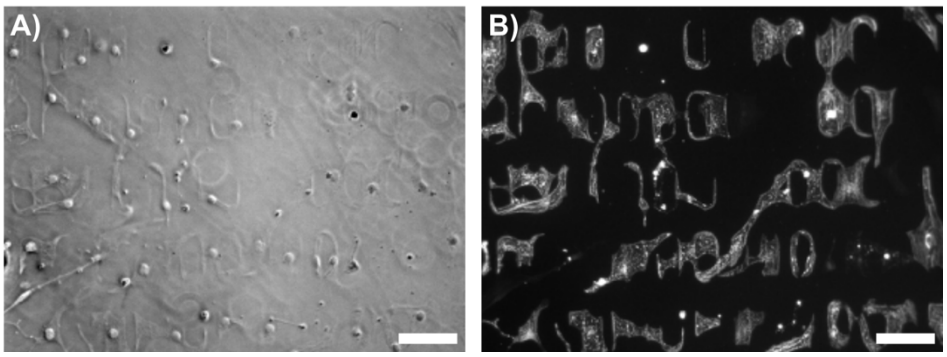

Figure 6-3: Conventional PDMS stamping on soft 15 kPa elastomer substrates. Stamp depth was 30 μm. Cells, here hiPSC-derived cardiomyocytes, were fixed and immunostained for α-actinin 2. **(A)** Bright-field image. **(B)** Epifluorescence image. Scale bars 70 μm. See text for discussion.

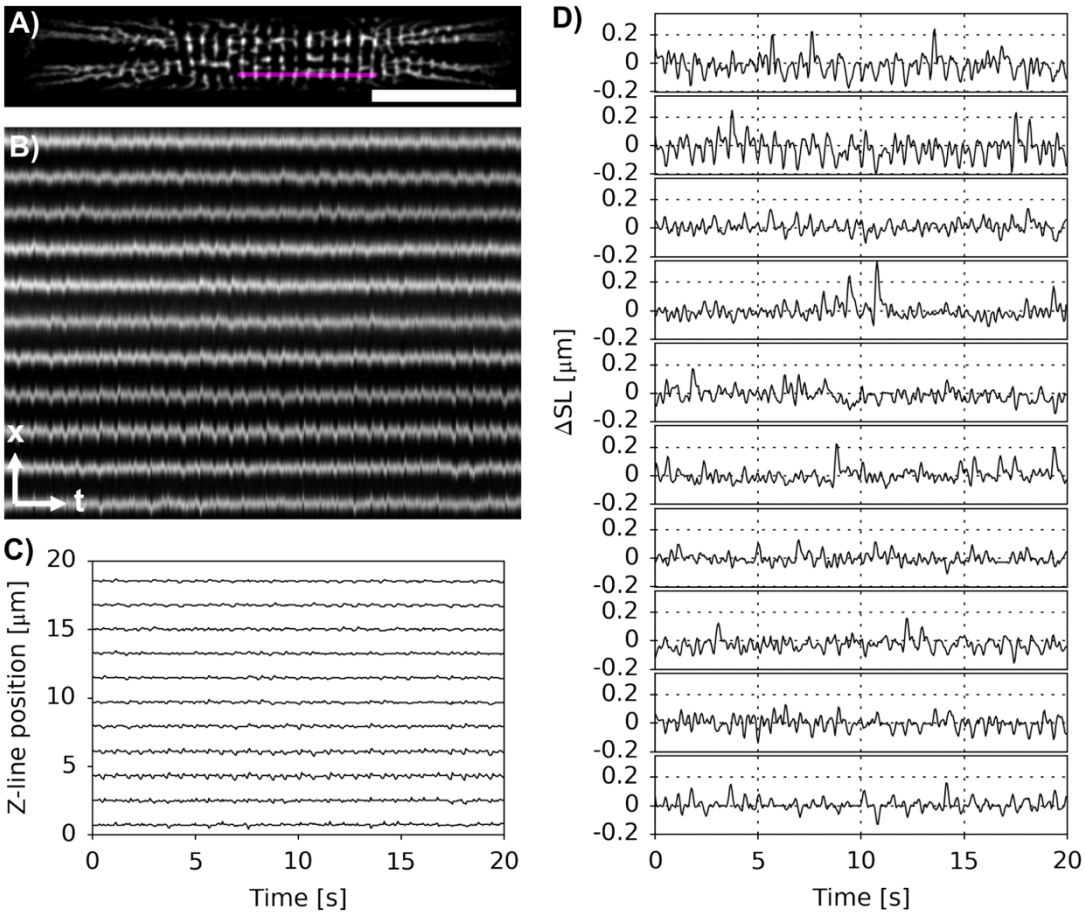

Figure 6-4: Representative HES-2-Citr-Linker-CM on glass. (A) Cardiomyocyte derived from HES-2-Citr-Linker on a micropatterned glass substrate. A myofibril, here comprising 10 sarcomeres, was selected as illustrated by the pink line. The intensity profile in the selected ROI was analyzed using our custom-designed algorithm to determine single Z-line trajectories. **(B)** Kymograph illustrating periodic displacement of Z-lines in the selected ROI. **(C)** Z-line trajectories, as determined via our custom-designed Python® script, mimic the above kymograph. **(D)** Change in sarcomere lengths, ΔSL, as derived from Z-line trajectories. Scale bar 20 µm.

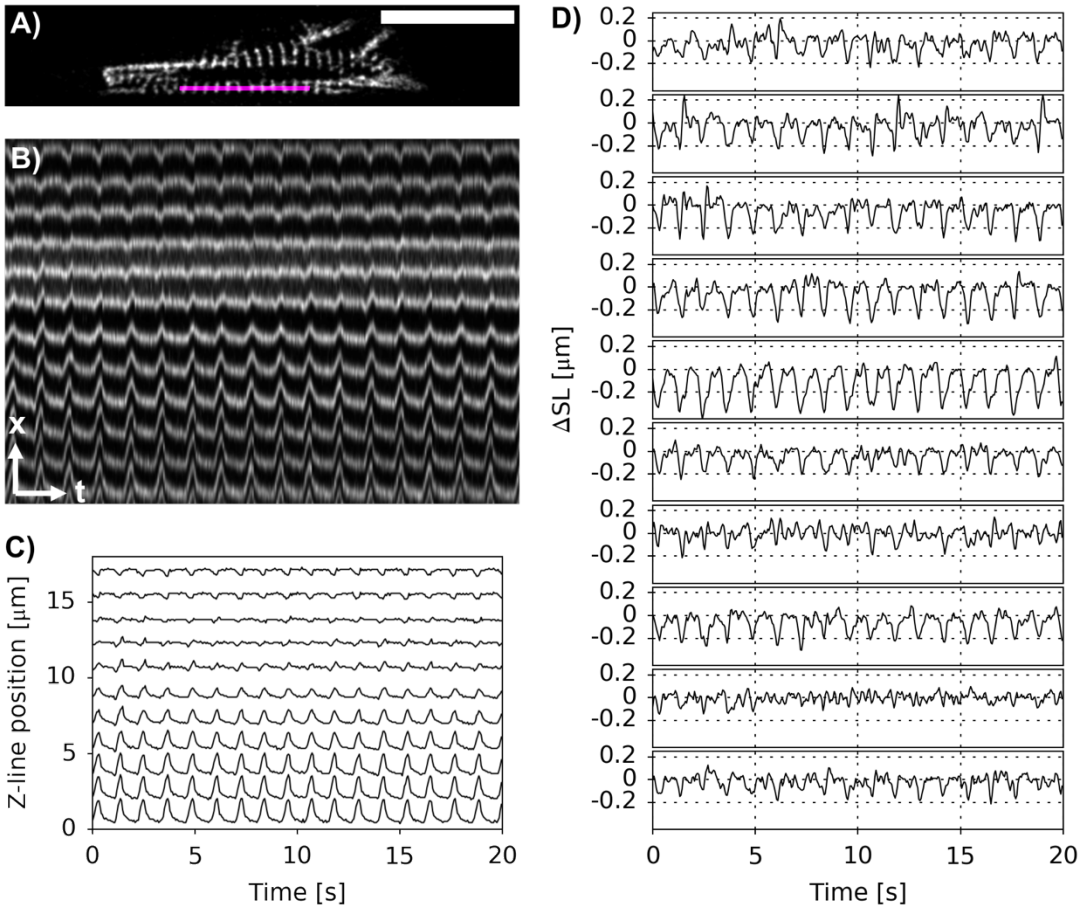

Figure 6-5: Representative HES-2-Citr-Linker-CM on a 15 kPa elastomer. (A) Cardiomyocyte derived from HES-2-Citr-Linker on a micropatterned 15 kPa elastomer substrate. A myofibril, here comprising 11 sarcomeres, was selected as illustrated by the pink line. The intensity profile in the selected ROI was analyzed using our custom-designed algorithm to determine single Z-line trajectories. **(B)** Kymograph illustrating periodic displacement of Z-lines in the selected ROI. **(C)** Z-line trajectories, as determined via our custom-designed Python® script, mimic the above kymograph. **(D)** Change in sarcomere lengths, ΔSL, as derived from Z-line trajectories. For concision purposes, only 10 sarcomeres are displayed in (D), and only the corresponding 11 Z-line trajectories are displayed in (C). Scale bar 20 µm.

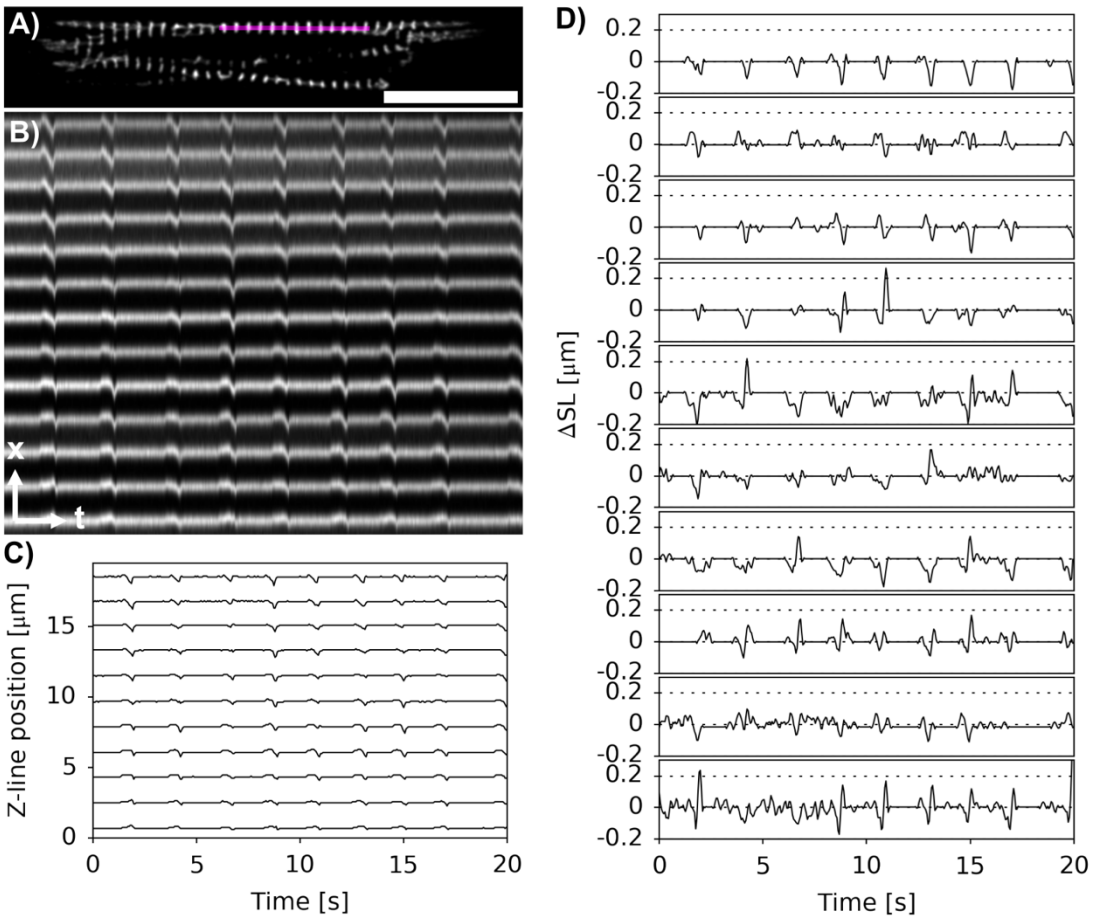

Figure 6-6: Representative HES-2-Citr-4Mut-CM on glass. (A) Cardiomyocyte derived from HES-2-Citr-4Mut on a micropatterned glass substrate. A myofibril, here comprising 12 sarcomeres, was selected as illustrated by the pink line. The intensity profile in the selected ROI was analyzed using our custom-designed algorithm to determine single Z-line trajectories. **(B)** Kymograph illustrating periodic displacement of Z-lines in the selected ROI. **(C)** Z-line trajectories, as determined via our custom-designed Python® script, mimic the above kymograph. **(D)** Change in sarcomere lengths, ΔSL, as derived from Z-line trajectories. For concision purposes, only 10 sarcomeres are displayed in (D), and only the corresponding 11 Z-line trajectories are displayed in (C). Scale bar 20 µm.

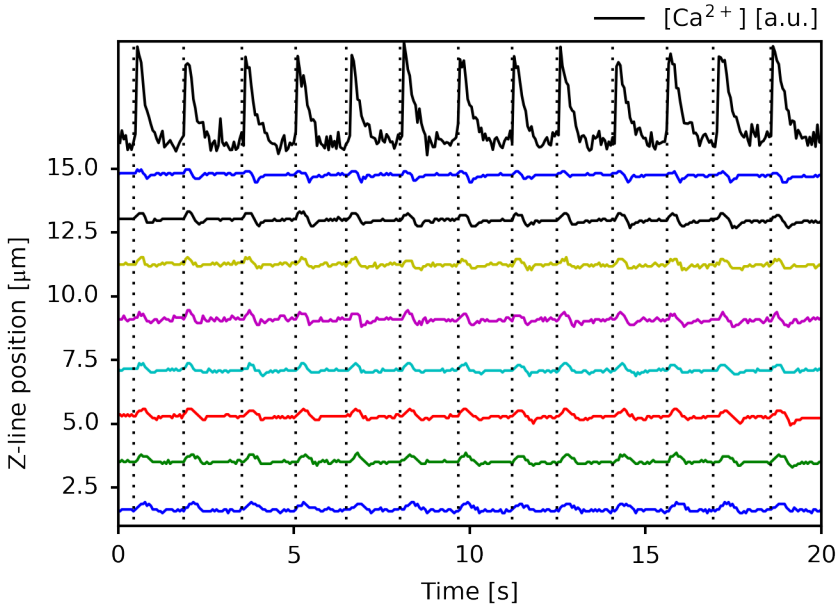

Figure 6-7: Calcium flux and Z-line trajectories of HES-2-ACTN2-Citr-CM. Z-line trajectories of HES-2-Citr-Linker-derviced CM on a micropatterned glass substrate (Figure 3-27). Initial displacement of Z-lines occurred at the increase in cytosolic calcium concentration.

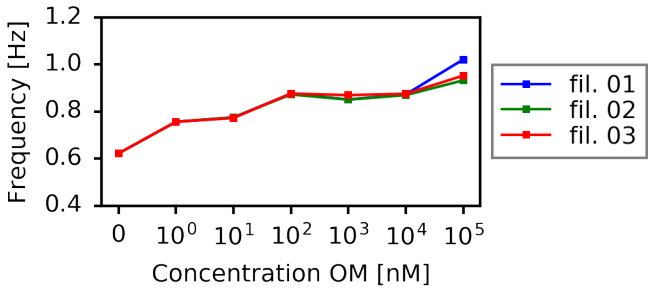

Figure 6-8: Concentration-frequency relationship for OM, as determined on glass. Three different myofibrils in the same single cell were analyzed. Concentration-frequency curves were plotted for each filament, individually. In the case of 100 µM OM, where most sarcomeres had entered into a state of constant contracture and where typical sarcomere contractions only occurred sparsely, the beating frequency was estimated from an overall undulation of the cell's Z-lines (see Figure 6-9). At all other OM concentrations, frequency was obtained via our analysis algorithm. ($p < 0.001$, as determined via rmANOVA.)

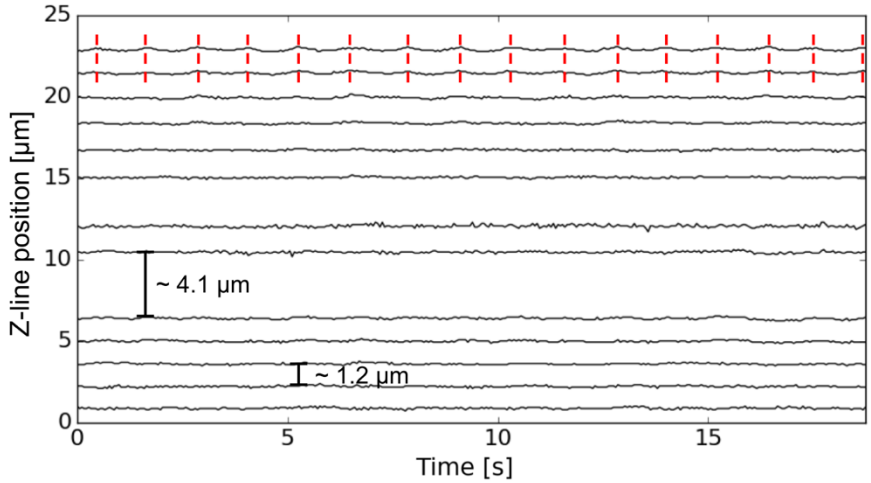

Figure 6-9: State of constant contracture. Graph shows position of Z-lines vs. time for a HES-2-Citr-4Mut-CM on 15 kPa treated with 10 µM omecamtiv mecarbil. At higher micromolar concentration of OM (here 10 µM), most sarcomeres seemed to enter into what appeared to be a state of constant contracture. While most sarcomeres were highly contracted (here as short as 1.2 µm), others seemed to get pulled apart by the (presumably stronger) neighboring sarcomeres (here as long as 4.1 µm). In this state of constant contracture, our analysis algorithm did not yield a beating frequency due to missing contractions. However, the "beating" frequency could still be estimated from periodic overall undulations of the cells' Z-lines/frame, which became particularly visible at the edge of the cells (see top two Z-lines). In this case, a total of 16 of these undulations occurred (as indicated by the red dashed lines) in 18.7 seconds, and thus f ~ 16/18.7 s = 0.86 Hz.

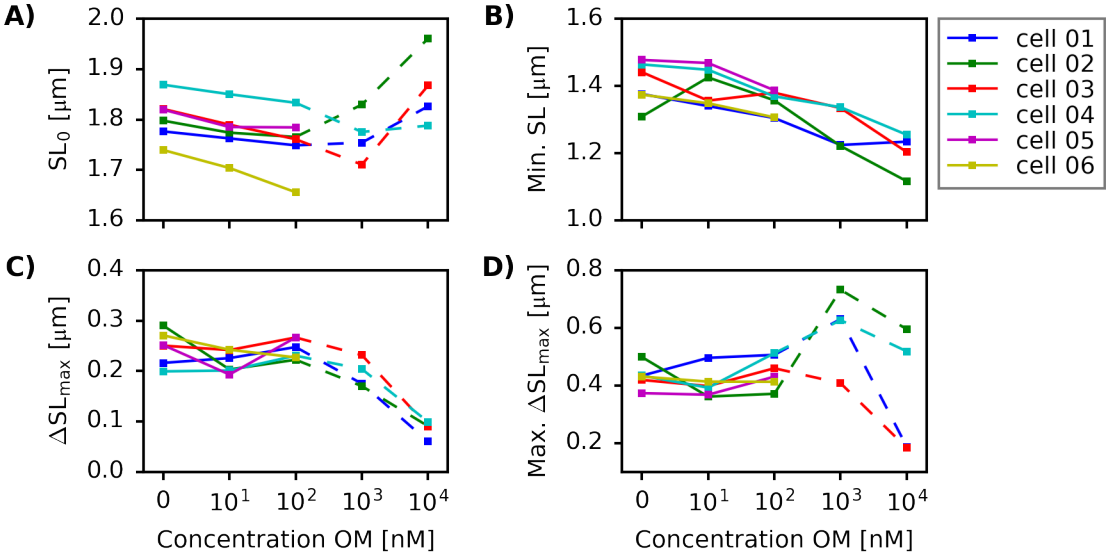

Figure 6-10: Effects of OM on sarcomere length and contraction amplitude. Concentration-response relationships on 15 kPa, as obtained for each cell individually. **(A)** Sarcomere length at rest. At micromolar concentrations, the average sarcomere rest length was highly biased due to ruptured sarcomeres with greater rest lengths as compared to intact sarcomeres. **(B)** Minimal sarcomere length observed during recording time. **(C)** Average contraction amplitude. **(D)** Maximal contraction amplitude observed during recording time. In (C) and (D), micromolar concentrations were excluded from analysis as the analysis algorithm was not applicable to determine ΔSL_{max} for uncoordinated beatings at 1 µM and at the state of constant contracture at 10 µM OM.

6.2 Relevant Nucleic Acid and Amino Acid Sequences

6.2.1 PCR Primer Sequences

Table 6-1: Primers used for amplification of the donor vectors.

Direction	Sequence (5' to 3')
Forward	CATGGGGTGAGGGTGGATGGG
Reverse	ACTGACTGAGTTGACCATCCAGC

Table 6-2: Primers used for genotyping and sequencing of transgenic cell lines.

Direction	Sequence (5' to 3')
Forward	GGAATTGTCCTATTTCCCACTG
Reverse	GCATGAAAATAAAACATTAGAATCC

6.2.2 Donor DNA Sequences

Table 6-3: Full sequences of donor DNA strands.

Donor Name	Sequence
SerGG	CATGGGGTGAGGGTGGATGGGAAATAGCCCAGCTGATTTAGAAGGTAAGAAACCATGGCTCCTCCAACC CCACTAAGAAACTTCTACTATAAATTATATAATATGCAAACTATAATGATATAAATTATAATGTGATAT TTGAGATTTACTTATTTTGACTTTTACCAACCAGACTATTTGGCTGGAATTGTCCTATTTCCCACTGAA CTTTTTTTTTAAAAGCTTCATCTTTTCTGGTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGG CTCAGTTTGAGAACTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGCCAG GCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTGTCTGGTACTACTATGCCAA TAGACTCCCTATTCTTTAGTCCTTTTAAAAAATTAAACAGATGCAAGAAATATGTAAGTATTAAGACTG TTTATGTTGTGGTGTTTCTGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGC GGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGGATGCCCGCCTACTC GGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGCGTTCTCTTCTGCACTCTACGGGGAGAGCGA TCTGGTGAGCAAGGGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGTCGAGCTGGACGGCGACGT AAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATGCCACCTACGGCAAGCTGACCCTGAA GTTCATCTGCACCACCGGCAAGCTGCCCGTGCCCTGGCCCACCCTCGTGACCACCTTCGGCTACGGCCT GATGTGCTTCGCCCGCTACCCCGACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGG CTACGTCCAGGAGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGACGGCAACATCCT GGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCATGGCCGACAAGCAGAAGAACGG CATCAAGGCCAACTTCAAGATCCGCCACAACATCGAGGACGGCGGCGTGCAGCTCGCCGACCACTACCA GCAGAACACCCCCATCGGCGACGGCCCCGTGCTGCTGCCCGACAACCACTACCTGAGCTACCAGTCCAA GCTGAGCAAAGACCCCAACGAGAAGCGCGATCACATGGTCCTGCTGGAGTTCGTGACCGCCGCCGGGAT CACTCTCGGCATGGACGAGCTATACAAGTGATGCTGAGCTTCTGTAATCACTCATCCCATCAGAATGCA ATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACAAGCTTTATTAAGTTGAGAGAGAGAGAG GGGAAAAAAAAAAGCCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACGGCTAAAATGAAGTTTTTA CAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAGCAAAATGTAATGAAATAT CAGGTTGATTTCTTTGATTAAACAGAACAAATTACTTGAGTAATAGGAAATTAGGAGGATCTAGGGACA GAAGGAAAGTGAAAAATGTGAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACCACAAAT TGGTAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGCTTATGTGATTGCCCCTA GGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGCTTATCCAAAGATTACTATTGTATCTTCAAA TGAATTTAATATTGTGAGATGGAACTGCCGGGGATTAAAAAGACTACCCAAAAGATTTTTGGCACTTAC AATTTTTAAAATAGTTTATGTCATCTCTTCATTATTTAGGGCTGGATGGTCAACTCAGTCAGT
4Mut	CATGGGGTGAGGGTGGATGGGAAATAGCCCAGCTGATTTAGAAGGTAAGAAACCATGGCTCCTCCAACC CCACTAAGAAACTTCTACTATAAATTATATAATATGCAAACTATAATGATATAAATTATAATGTGATAT TTGAGATTTACTTATTTTGACTTTTACCAACCAGACTATTTGGCTGGAATTGTCCTATTTCCCACTGAA CTTTTTTTTTAAAAGCTTCATCTTTTCTGGTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGG CTCAGTTTGAGAACTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGCCAG GCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTGTCTGGTACTACTATGCCAA TAGACTCCCTATTCTTTAGTCCTTTTAAAAAATTAAACAGATGCAAGAAATATGTAAGTATTAAGACTG TTTATGTTGTGGTGTTTCTGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGC GGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGGATGCCCGCCTACTC GGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGCGTTCTCTTCTGCTTTGTATGGGGAGAGCGA TCTGGTGAGCAAGGGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGTCGAGCTGGACGGCGACGT AAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATGCCACCTACGGCAAGCTGACCCTGAA GTTCATCTGCACCACCGGCAAGCTGCCCGTGCCCTGGCCCACCCTCGTGACCACCTTCGGCTACGGCCT GATGTGCTTCGCCCGCTACCCCGACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGG CTACGTCCAGGAGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGACGGCAACATCCT GGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCATGGCCGACAAGCAGAAGAACGG CATCAAGGCCAACTTCAAGATCCGCCACAACATCGAGGACGGCGGCGTGCAGCTCGCCGACCACTACCA GCAGAACACCCCCATCGGCGACGGCCCCGTGCTGCTGCCCGACAACCACTACCTGAGCTACCAGTCCAA GCTGAGCAAAGACCCCAACGAGAAGCGCGATCACATGGTCCTGCTGGAGTTCGTGACCGCCGCCGGGAT CACTCTCGGCATGGACGAGCTATACAAGTGATGCTGAGCTTCTGTAATCACTCATCCCATCAGAATGCA ATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACAAGCTTTATTAAGTTGAGAGAGAGAGAG

GGGAAAAAAAAAAGCCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACGGCTAAAATGAAGTTTTTA
CAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAGCAAAATGTAATGAAATAT
CAGGTTGATTTCTTTGATTAAACAGAACAAATTACTTGAGTAATAGGAAATTAGGAGGATCTAGGGACA
GAAGGAAAGTGAAAAATGTGAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACCACAAAT
TGGTAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGCTTATGTGATTGCCCCTA
GGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGCTTATCCAAAGATTACTATTGTATCTTCAAA
TGAATTTAATATTGTGAGATGGAACTGCCGGGGATTAAAAAGACTACCCAAAAGATTTTTGGCACTTAC
AATTTTTAAAATAGTTTATGTCATCTCTTCATTATTTAGGGCTGGATGGTCAACTCAGTCAGT

Linker

CATGGGGTGAGGGTGGATGGGAAATAGCCCAGCTGATTTAGAAGGTAAGAAACCATGGCTCCTCCAACC
CCACTAAGAAACTTCTACTATAAATTATATAATATGCAAACTATAATGATATAAATTATAATGTGATAT
TTGAGATTTACTTATTTTGACTTTTACCAACCAGACTATTTGGCTGGAATTGTCCTATTTCCCACTGAA
CTTTTTTTTTAAAAGCTTCATCTTTTCTGGTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGG
CTCAGTTTGAGAACTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGCCAG
GCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTGTCTGGTACTACTATGCCAA
TAGACTCCCTATTCTTTAGTCCTTTTAAAAAATTAAACAGATGCAAGAAATATGTAAGTATTAAGACTG
TTTATGTTGTGGTGTTTCTGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGC
GGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGGATGCCCGCCTACTC
GGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGCGTTCTCTTCTGCTTTGTATGGGGAGAGCGA
TCTGGGAGGTGGAGGTGGAGCTGTGAGCAAGGGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGT
CGAGCTGGACGGCGACGTAAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATGCCACCTA
CGGCAAGCTGACCCTGAAGTTCATCTGCACCACCGGCAAGCTGCCCGTGCCCTGGCCCACCCTCGTGAC
CACCTTCGGCTACGGCCTGATGTGCTTCGCCCGCTACCCCGACCACATGAAGCAGCACGACTTCTTCAA
GTCCGCCATGCCCGAAGGCTACGTCCAGGAGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGAC
CCGCGCCGAGGTGAAGTTCGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAA
GGAGGACGGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCATGGC
CGACAAGCAGAAGAACGGCATCAAGGCCAACTTCAAGATCCGCCACAACATCGAGGACGGCGGCGTGCA
GCTCGCCGACCACTACCAGCAGAACACCCCCATCGGCGACGGCCCCGTGCTGCTGCCCGACAACCACTA
CCTGAGCTACCAGTCCAAGCTGAGCAAAGACCCCAACGAGAAGCGCGATCACATGGTCCTGCTGGAGTT
CGTGACCGCCGCCGGGATCACTCTCGGCATGGACGAGCTATACAAGTGATGCTGAGCTTCTGTAATCAC
TCATCCCATCAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACAAGCTTTATT
AAGTTGAGAGAGAGAGAGGGGAAAAAAAAAAGCCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACG
GCTAAAATGAAGTTTTTACAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAG
CAAAATGTAATGAAATATCAGGTTGATTTCTTTGATTAAACAGAACAAATTACTTGAGTAATAGGAAAT
TAGGAGGATCTAGGGACAGAAGGAAAGTGAAAAATGTGAAAATACAAAATACCCAAGATTTAAGACCGG
GGGGAAAAAACCACAAATTGGTAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATG
CTTATGTGATTGCCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGCTTATCCAAAGAT
TACTATTGTATCTTCAAATGAATTTAATATTGTGAGATGGAACTGCCGGGGATTAAAAAGACTACCCAA
AAGATTTTTGGCACTTACAATTTTTAAAATAGTTTATGTCATCTCTTCATTATTTAGGGCTGGATGGTC
AACTCAGTCAGT

Lightly red highlighted regions are part of the exon 21 of ACTN2. Written in green are the binding domain sequence and respective PAM sequence of gRNA-2. Highlighted in grey are the binding domain sequence and respective PAM sequence of gRNA-1. Highlighted in pink are silent mutations, which were introduced to inhibit cleavage of the donor strands. Yellow highlighting denotes the coding sequence for Citrine YFP, and highlighted in green is the sequence encoding for a small linker peptide $(Gly)_5$-Ala, which was inserted in the case of the donor termed *Linker*. Highlighted in red is the endogenous STOP codon.

6.2.3 DNA and Amino Acid Sequences of Transgenic Cell Lines

Green highlighting marks DNA amplification primers or linker sequence

Yellow highlighting marks Citrine sequence

CCA marks start of last ACTN2 exon

TGA marks endogenous STOP codon of last ACTN2 exon

T marks Cas9 cleaving site

T marks base pair insertion in untargeted alleles

TAG marks shifted STOP codon resulting from indel in untargeted alleles

Grey highlighting marks point mutations in the 5' and 3' UTR that only occurred in some of the multiple sequencing attempts. We did not observe these mutations in other attempts (data not shown), and thus these mutations could be attributed to minor PCR errors.

Untargeted and WT alleles

```
Endogenous_ACTN2          GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTG
WT_M13_rev                GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTG
4Mut_untargeted_M13_rev   GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTG
Linker_untargeted_T7_rev  GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTG

Endogenous_ACTN2          GTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAA
WT_M13_rev                GTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAA
4Mut_untargeted_M13_rev   GTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAA
Linker_untargeted_T7_rev  GTATGAAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAA

Endogenous_ACTN2          CTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGC
WT_M13_rev                CTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGC
4Mut_untargeted_M13_rev   CTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGC
Linker_untargeted_T7_rev  CTACTAGTAATGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGC

Endogenous_ACTN2          CAGGCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTG
WT_M13_rev                CAGGCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTG
4Mut_untargeted_M13_rev   CAGGCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTG
Linker_untargeted_T7_rev  CAGGCCGCTTCTAGATACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTG

Endogenous_ACTN2          TCTGGTACTACTATGCCAATAGACTCCCTATTCTTTAGTCCTTTTAAAAAATT
WT_M13_rev                TCTGGTACTACTATGCCAATAGACTCCCTATTCTTTAGTCCTTTTAAAAAATT
4Mut_untargeted_M13_rev   TCTGGTACTACTATGCCAATAGACTCCCTATTCTTTAGTCCTTTTAAAAAATT
Linker_untargeted_T7_rev  TCTGGTACTACTATGCCAATAGACTCCCTATTCTTTAGTCCTTTTAAAAAATT

Endogenous_ACTN2          AAACAGATGCAAGAAATATGTAAGTATTAAGACTGTTTATGTTGTGGTGTTTC
WT_M13_rev                AAACAGATGCAAGAAATATGTAAGTATTAAGACTGTTTATGTTGTGGTGTTTC
4Mut_untargeted_M13_rev   AAACAGATGCAAGAAATATGTAAGTATTAAGACTGTTTATGTTGTGGTGTTTC
Linker_untargeted_T7_rev  AAACAGATGCAAGAAATATGTAAGTATTAAGACTGTTTATGTTGTGGTGTTTC

Endogenous_ACTN2          TGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGCGG
WT_M13_rev                TGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGCGG
4Mut_untargeted_M13_rev   TGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGCGG
Linker_untargeted_T7_rev  TGCAACTGACTGCAAACACGTGTGTATTTTTTCCCAGCCATACATCCTGGCGG

Endogenous_ACTN2          AGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGG
WT_M13_rev                AGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGG
4Mut_untargeted_M13_rev   AGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGG
Linker_untargeted_T7_rev  AGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGCATCAAGAGG

Endogenous_ACTN2          ATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGC
WT_M13_rev                ATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGC
4Mut_untargeted_M13_rev   ATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGC
Linker_untargeted_T7_rev  ATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCTGC

Endogenous_ACTN2          GTTCTCTTCCGCACT-CTACGGGGAGAGCGATCTGTGATGCTGAGCTTCTGTA
WT_M13_rev                GTTCTCTTCCGCACT-CTACGGGGAGAGCGATCTGTGATGCTGAGCTTCTGTA
4Mut_untargeted_M13_rev   GTTCTCTTCCGCACTTCTACGGGGAGAGCGATCTGTGATGCTGAGCTTCTGTA
Linker_untargeted_T7_rev  GTTCTCTTCCGCACTTCTACGGGGAGAGCGATCTGTGATGCTGAGCTTCTGTA
```

```
Endogenous_ACTN2          ATCACTCATCCCATCAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTG
WT_M13_rev                ATCACTCATCCCATCAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTG
4Mut_untargeted_M13_rev   ATCACTCATCCCATCAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTG
Linker_untargeted_T7_rev  ATCACTCATCCCATCAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTG

Endogenous_ACTN2          GAAACTTTGACAAGCTTTATTAAGTTGAGAGAGAGAGAGGGGAAAAAAAAAAG
WT_M13_rev                GAAACTTTGACAAGCTTTATTAAGTTGAGAGAGAGAGAGGGGGAAAAAAAAAG
4Mut_untargeted_M13_rev   GAAACTTTGACAAGCTTTATTAAGTTGAGAGAGAGAGAGGGGGAAAAAAAAAG
Linker_untargeted_T7_rev  GAAACTTTGACAAGCTTTATTAAGTTGAGAGAGAGAGAGGGGGAAAAAAAAAG

Endogenous_ACTN2          CCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACGGCTAAAATGAAGTTTT
WT_M13_rev                CCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACGGCTAAAATGAAGTTTT
4Mut_untargeted_M13_rev   CCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACGGCTAAAATGAAGTTTT
Linker_untargeted_T7_rev  CCTTTCGTAGTTCAGTAATTGCCAGCAATATAACACGGCTAAAATGAAGTTTT

Endogenous_ACTN2          TACAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAG
WT_M13_rev                TACAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAG
4Mut_untargeted_M13_rev   TACAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAG
Linker_untargeted_T7_rev  TACAGTATATGACATAGTGCGCTTCATAAATAGGTTTATTTCTGAGTTTTTAG

Endogenous_ACTN2          CAAAATGTAATGAAATATCAGGTTGATTTCTTTGATTAAACAGAACAAATTAC
WT_M13_rev                CAAAATGTAATGAAATATCAGGTTGATTTCTTTGATTAAACAGAACAAATTAC
4Mut_untargeted_M13_rev   CAAAATGTAATGAAATATCAGGTTGATTTCTTTGATTAAACAGAACAAATTAC
Linker_untargeted_T7_rev  CAAAATGTAATGAAATATCAGGTTGATTTCTTTGATTAAACAGAACAAATTAC

Endogenous_ACTN2          TTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGGAAAGTGAAAAATGT
WT_M13_rev                TTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGGAAAGTGAAAAATGT
4Mut_untargeted_M13_rev   TTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGGAAAGTGAAAAATGT
Linker_untargeted_T7_rev  TTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGGAAAGTGAAAAATGT

Endogenous_ACTN2          GAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACCACAAATTGG
WT_M13_rev                GAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACCACAAATTGG
4Mut_untargeted_M13_rev   GAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACCACAAATTGG
Linker_untargeted_T7_rev  GAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACCACAAATTGG

Endogenous_ACTN2          TAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGCTTAT
WT_M13_rev                TAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGCTTAT
4Mut_untargeted_M13_rev   TAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGCTTAT
Linker_untargeted_T7_rev  TAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGCTTAT

Endogenous_ACTN2          GTGATTGCCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
WT_M13_rev                GTGATTGGCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
4Mut_untargeted_M13_rev   GTGATTGCCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
Linker_untargeted_T7_rev  GTGATTGGCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
```

Knock-in alleles

```
Linker_target_sequence    GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTGGTATG
4Mut_knock-in_for         GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTGGTATG
Linker_knock-in_for       GGAATTGTCCTATTTCCCACTGAACTTTTTTTTTAAAAGCTTCATCTTTTCTGGTATG

Linker_target_sequence    AAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAACTACTAGTAA
4Mut_knock-in_for         AAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAACTACTAGTAA
Linker_knock-in_for       AAATGCAGATCATAGTACGTATCCTCGCATTGTGGCTCAGTTTGAGAACTACTAGTAA

Linker_target_sequence    TGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGCCAGGCCGCTTCTAGA
4Mut_knock-in_for         TGCTCCATTTGCCTTTATGAAGCATATCACCCTTCCTGCTGGCCAGGCCGCTTCTAGA
Linker_knock-in_for       TGCTCCATTTGCCTTTATGAAGCATATCACCCCTCCTGCTGGCCAGGCCGCTTCTAGA

Linker_target_sequence    TACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTGTCTGGTACTACTATGCCAAT
4Mut_knock-in_for         TACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTGTCTGGTACTACTATGCCAAT
Linker_knock-in_for       TACGCTCCTACAAGTAAAACTTGGCTTTCTGTTGGTTGTCTGGTACTACTATGCCAAT

Linker_target_sequence    AGACTCCCTATTCTTTAGTCCTTTTAAAAAATTAAACAGATGCAAGAAATATGTAAGT
4Mut_knock-in_for         AGACTCCCTATTCTTTAGTCCTTTTAAAAAATTAAACAGATGCAAGAAACATGTAAGT
Linker_knock-in_for       AGACTCCCTATTCTTTAGTCCTTTTAAAAAATTAAACAGATGCAAGAAATATGTAAGT

Linker_target_sequence    ATTAAGACTGTTTATGTTGTGGTGTTTCTGCAACTGACTGCAAACACGTGTGTATTTT
4Mut_knock-in_for         ATTAAGACTGTTTATGTTGTGGTGTTTCTGCAACTGACTGCAAACACGTGTGTATTTT
Linker_knock-in_for       ATTAAGACTGTTTATGTTGTGGTGTTTCTGCAACTGACTGCAAACACGTGTGTATTTT

Linker_target_sequence    TTCCCAGCCATACATCCTGGCGGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCC
4Mut_knock-in_for         TTCCCAGCCATACATCCTGGCGGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCC
Linker_knock-in_for       TTCCCAGCCATACATCCTGGCGGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCC
```

```
Linker_target_sequence    CAGTACTGCATCAAGAGGATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCAC
4Mut_knock-in_for         CAGTACTGCATCAAGAGGATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCAC
Linker_knock-in_for       CAGTACTGCATCAAGAGGATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCAC

Linker_target_sequence    TGGATTACGCTGCGTTCTCTTCTGCTTTGTATGGGGAGAGCGATCTGGGAGGTGGAGG
4Mut_knock-in_for         TGGATTACGCTGCGTTCTCTTCTGCTTTGTATGGGGAGAGCGATCTG----------
Linker_knock-in_for       TGGATTACGCTGCGTTCTCTTCTGCTTTGTATGGGGAGAGCGATCTGGGAGGTGGAGG

Linker_target_sequence    TGGAGCTGTGAGCAAGGGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGTCGAG
4Mut_knock-in_for         -------GTGAGCAAGGGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGTCGAG
Linker_knock-in_for       TGGAGCTGTGAGCAAGGGCGAGGAGCTGTTCACCGGGGTGGTGCCCATCCTGGTCGAG

Linker_target_sequence    CTGGACGGCGACGTAAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATG
4Mut_knock-in_for         CTGGACGGCGACGTAAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATG
Linker_knock-in_for       CTGGACGGCGGCGTAAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGAGGGCGATG

Linker_target_sequence    CCACCTACGGCAAGCTGACCCTGAAGTTCATCTGCACCACCGGCAAGCTGCCCGTGCC
4Mut_knock-in_for         CCACCTACGGCAAGCTGACCCTGAAGTTCATCTGCACCACCGGCAAGCTGCCCGTGCC
Linker_knock-in_for       CCACCTACGGCAAGCTGACCCTGAAGTTCATCTGCACCACCGGCAAGCTGCCCGTGCC
4Mut_knock-in_rev                                     ATTTGCACCACCGGCAAGCTGCCCGTGCC
Linker_knock-in_rev               GCAAGMTGACCCTGAAGTTCATTTGCACCACCGGCAAGCTGCCCGTGCC

Linker_target_sequence    CTGGCCCACCCTCGTGACCACCTTCGGCTACGGCCTGATGTGCTTCGCCCGCTACCCC
4Mut_knock-in_for         CTGGCCCACCCTCGTGACCACCTTCGGCTACGGCCTGATGTGCTTCGCCCGCTACCCC
Linker_knock-in_for       CTGGCCCACCCTCGTGACCACCTTCGGCTACGGCCTGATGTGCTTCGCCCGCTACCCC
4Mut_knock-in_rev         CTGGCCCACCCTCGTGACCACCTTTGGGTTACGGCCTGATGTGCTTCGCCCGCTACCCC
Linker_knock-in_rev       CTGGCCCACCCTCGTGACCACTTTGGGTTACGGCCTGATGTGCTTCGCCCGCTACCCC

Linker_target_sequence    GACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGGCTACGTCCAGG
4Mut_knock-in_for         GACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGGCTACGTCCAGG
Linker_knock-in_for       GACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGGCTACGTCCAGG
4Mut_knock-in_rev         GACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGGCTACGTCCAGG
Linker_knock-in_rev       GACCACATGAAGCAGCACGACTTCTTCAAGTCCGCCATGCCCGAAGGCTACGTCCAGG

Linker_target_sequence    AGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT
4Mut_knock-in_for         AGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT
Linker_knock-in_for       AGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT
4Mut_knock-in_rev         AGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT
Linker_knock-in_rev       AGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGAGGTGAAGTT

Linker_target_sequence    CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGAC
4Mut_knock-in_for         CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGAC
Linker_knock-in_for       CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGAC
4Mut_knock-in_rev         CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGAC
Linker_knock-in_rev       CGAGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGAC

Linker_target_sequence    GGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCA
4Mut_knock-in_for         GGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCA
Linker_knock-in_for       GGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCA
4Mut_knock-in_rev         GGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCA
Linker_knock-in_rev       GGCAACATCCTGGGGCACAAGCTGGAGTACAACTACAACAGCCACAACGTCTATATCA

Linker_target_sequence    TGGCCGACAAGCAGAAGAACGGCATCAAGGCCAACTTCAAGATCCGCCACAACATCGA
4Mut_knock-in_for         TGGCCGACAAGCAGAAGAACGGCATCAAGGCCAACTTCAAGATCCGCCACAACATCGA
Linker_knock-in_for       TGGCCGACAAGCAGAAGAACGGCATCAAGGCCAACTTCAAGATCCGCCACAACATCGA
4Mut_knock-in_rev         TGGCCGACAAGCAGAAGAACGGCATCAAGGCCAACTTCAAGATCCGCCACAACATCGA
Linker_knock-in_rev       TGGCCGACAAGCAGAAGAACGGCATCAAGGCCAACTTCAAGATCCGCCACAACATCGA

Linker_target_sequence    GGA-CGGCGGCGTGCAGCTCGCCGACCACTACCAGCAGAACACCCCCATCGGCGACGG
4Mut_knock-in_for         GGAACGGCGGCGTGCAGCTCGCCGACCACTACCAGCAGAACACCCCCMTCGGGGAAGG
Linker_knock-in_for       GGA-CGGCGGCGTGCAGCTCGCCGACCACTACCAGCAGAACACCCCCATCGGCGACGG
4Mut_knock-in_rev         GGA-CGGCGGCGTGCAGCTCGCCGACCACTACCAGCAGAACACCCCCATCGGCGACGG
Linker_knock-in_rev       GGA-CGGCGGCGTGCAGCTCGCCGACCACTACCAGCAGAACACCCCCATCGGCGACGG

Linker_target_sequence    CCCCGTGCTGCTGCCCGACAACCACTACCTGAGCTACCAGTCCAAGCTGAGCAAAGAC
Linker_knock-in_for       CCCCGTGCTGCTGCCCSAAAACCACTACCTGAGCTACCAGTCCAAGCTGAACAAAGAC
4Mut_knock-in_rev         CCCCGTGCTGCTGCCCGACAACCACTACCTGAGCTACCAGTCCAAGCTGAGCAAAGAC
Linker_knock-in_rev       CCCCGTGCTGCTGCCCGACAACCACTACCTGAGCTACCAGTCCAAGCTGAGCAAAGAC

Linker_target_sequence    CCCAACGAGAAGCGCGATCACATGGTCCTGCTGGAGTTCGTGACCGCCGCCGGGATCA
Linker_knock-in_for       CCCAACG
4Mut_knock-in_rev         CCCAACGAGAAGCGCGATCACATGGTCCTGCTGGAGTTCGTGACCGCCGCCGGGATCA
Linker_knock-in_rev       CCCAACGAGAAGCGCGATCACATGGTCCTGCTGGAGTTCGTGACCGCCGCCGGGATCA

Linker_target_sequence    CTCTCGGCATGGACGAGCTATACAAGTGATGCTGAGCTTCTGTAATCACTCATCCCAT
```

```
4Mut_knock-in_rev          CTCTCGGCATGGACGAGCTATACAAGTGATGCTGAGCTTCTGTAATCACTCATCCCAT
Linker_knock-in_rev        CTCTCGGCATGGACGAGCTATACAAGTGATGCTGAGCTTCTGTAATCACTCATCCCAT

Linker_target_sequence     CAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACAAGCTTTA
4Mut_knock-in_rev          CAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACAAGCTTTA
Linker_knock-in_rev        CAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACAAGCTTTA

Linker_target_sequence     TTAAGTTGAGAGAGAGAGAGGGGAAAAAAAAAAAGCCTTTCGTAGTTCAGTAATTGCCA
4Mut_knock-in_rev          TTAAGTTGAGAGAGAGAGAGGGGAAAAAAAAAAAGCCTTTCGTAGTTCAGTAATTGCCA
Linker_knock-in_rev        TTAAGTTGAGAGAGAGAGAGGGGAAAAAAAAAAAGCCTTTCGTAGTTCAGTAATTGCCA

Linker_target_sequence     GCAATATAACACGGCTAAAATGAAGTTTTTACAGTATATGACATAGTGCGCTTCATAA
4Mut_knock-in_rev          GCAATATAACACGGCTAAAATGAAGTTTTTACAGTATATGACATAGTGCGCTTCATAA
Linker_knock-in_rev        GCAATATAACACGGCTAAAATGAAGTTTTTACAGTATATGACATAGTGCGCTTCATAA

Linker_target_sequence     ATAGGTTTATTTCTGAGTTTTTAGCAAAATGTAATGAAATATCAGGTTGATTTCTTTG
4Mut_knock-in_rev          ATAGGTTTATTTCTGAGTTTTTAGCAAAATGTAATGAAATATCAGGTTGATTTCTTTG
Linker_knock-in_rev        ATAGGTTTATTTCTGAGTTTTTAGCAAAATGTAATGAAATATCAGGTTGATTTCTTTG

Linker_target_sequence     ATTAAACAGAACAAATTACTTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGG
4Mut_knock-in_rev          ATTAAACAGAACAAATTACTTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGG
Linker_knock-in_rev        ATTAAACAGAACAAATTACTTGAGTAATAGGAAATTAGGAGGATCTAGGGACAGAAGG

Linker_target_sequence     AAAGTGAAAAATGTGAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACC
4Mut_knock-in_rev          AAAGTGAAAAATGTGAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACC
Linker_knock-in_rev        AAAGTGAAAAATGTGAAAATACAAAATACCCAAGATTTAAGACCGGGGGGAAAAAACC

Linker_target_sequence     ACAAATTGGTAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGC
4Mut_knock-in_rev          ACAAATTGGTAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGC
Linker_knock-in_rev        ACAAATTGGTAAATAAAGGTTTGCTATTTGTAAAAAATTTCATTTATCTCTAATATGC

Linker_target_sequence     TTATGTGATTGCCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
4Mut_knock-in_rev          TTATGTGATTGCCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
Linker_knock-in_rev        TTATGTGATTGCCCCTAGGGGAGTATATTTGGGATTCTAATGTTTTATTTTCATGC
```

WT amino acid sequence of the last ACTN2 exon

```
CCATACATCCTGGCGGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGC
 P  Y  I  L  A  E  E  L  R  R  E  L  P  P  D  Q  A  Q  Y  C
ATCAAGAGGATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCT
 I  K  R Met P  A  Y  S  G  P  G  S  V  P  G  A  L  D  Y  A
GCGTTCTCTTCCGCACTCTACGGGGAGAGCGATCTGTGA
 A  F  S  S  A  L  Y  G  E  S  D  L  -
```

Resulting Amino Acid Sequence of the mutated WT allele in the untargeted knock-in alleles

```
CCATACATCCTGGCGGAGGAGCTGCGTCGGGAGCTGCCCCCGGATCAGGCCCAGTACTGC
 P  Y  I  L  A  E  E  L  R  R  E  L  P  P  D  Q  A  Q  Y  C
ATCAAGAGGATGCCCGCCTACTCGGGCCCAGGCAGTGTGCCTGGTGCACTGGATTACGCT
 I  K  R Met P  A  Y  S  G  P  G  S  V  P  G  A  L  D  Y  A
GCGTTCTCTTCCGCACTTCTACGGGGAGAGCGATCTGTGATGCTGAGCTTCTGTAATCAC
 A  F  S  S  A  L  L  R  G  E  R  S  V Met L  S  F  C  N  H
TCATCCCATCAGAATGCAATAAAAGCGGAAGTCACAGTTTGTTTCCTGGAAACTTTGACA
 S  S  H  Q  N  A  I  K  A  E  V  T  V  C  F  L  E  T  L  T
AGCTTTATTAAGTTGAGAGAGAGAGAGGGGAAAAAAAAAAGCCTTTCGTAG
 S  F  I  K  L  R  E  R  E  G  K  K  K  S  L  S  -
```

6.3 Reagents and Media

hES medium

- 385 ml of Knockout Dulbecco's modified Eagle's medium (DMEM, Cat 10829, Invitrogen, Carlsbad, USA)
- 100 ml of Knockout serum replacement (Cat 10828, Invitrogen, Carlsbad, USA)
- 5 ml of non-essential amino acids (final conc.: 0.1 mM; Cat 11140, Invitrogen, Carlsbad, USA)
- 5 ml of l-glutamine (final conc.: 2 mM; Cat 25030, Invitrogen, Carlsbad, USA)
- 5 ml of penicillin/streptomycin (final conc.: 100 U/ml/100 µg/ml; Cat 15140, Invitrogen, Carlsbad, USA)
- 10 ng/ml basic fibroblast growth factor, typically added to smaller sized aliquots of hES medium (Cat 130-093-841, Miltenyi Biotech, Bergisch Gladbach, Germany)

E8 medium

- TeSR™-E8™ medium prepared from TeSR™-E8™ kit according to the manufacturer's protocol (Cat 05940, Stemcell Technologies, Vancouver, Canada)
- 100 U/ml penicillin/100 µg/ml streptomycin (Cat 15140, Invitrogen, Carlsbad, USA)

RPMI basal medium

- 480 ml of Roswell Park Memorial Institute medium (RPMI) 1640 with Glutamax (Cat 61870-010, Invitrogen, Carlsbad, USA)
- 5 ml of sodium pyruvate (final conc.: 1 mM; Cat 11360, Invitrogen, Carlsbad, USA)
- 5 ml of penicillin/streptomycin (final conc.: 100 U/ml/100 µg/ml; Cat 15140, Invitrogen, Carlsbad, USA)
- 10 ml of B27 supplement (Cat 17504-044, Invitrogen, Carlsbad, USA)
- 200 µM L-ascorbic acid (Cat A8960-5G, Sigma Aldrich, Taufkirchen, Germany)

Mesodermal induction medium

- RPMI basal medium (prewarmed)
- 1 µM CHIR (Cat 04-0004, Stemgent, Cambridge, USA)
- 5 ng/ml bone morphogenetic protein 4 (BMP4, Cat 314-BP, R&D systems, Minneapolis, USA)
- 9 ng/ml activin A (Cat 338-AC, R&D systems, Minneapolis, USA)
- 5 ng/ml basic fibroblast growth factor (bFGF, Cat 130-093-841, Stemgent, Cambridge, USA)

Cardiac differentiation medium

- RPMI basal medium (prewarmed)
- 5 µM IWP4 (Cat 04-0036, Stemgent, Cambridge, USA)

Selection medium

- 49.15 ml of glucose-free Roswell Park Memorial Institute medium (RPMI) 1640 (Cat 11879020, Life Technologies/Thermo Fisher Scientific, Waltham, USA)
- 0.5 ml of penicillin/streptomycin (final conc.: 100 U/ml/100 µg/ml; Cat 15140, Invitrogen, Carlsbad, USA)
- 0.25 ml of sodium lactate (final conc.: 2.2 mM; Cat 71723, Sigma Aldrich, Taufkirchen, Germany)
- 0.1 ml of 2-Mercaptoethanol (final conc.: 0.1 mM; Cat 31350010, Invitrogen, Carlsbad, USA)

DMEM, 0.8 mM Ca^{2+}

- 485 ml of Dulbecco's modified Eagle's medium with 4.5 g/l D-glucose (DMEM, Cat 21068-028, Invitrogen, Carlsbad, USA)
- 5 ml of sodium pyruvate (final conc.: 1 mM; Cat 11360, Invitrogen, Carlsbad, USA)
- 5 ml of l-glutamine (final conc.: 2 mM; Cat 25030, Invitrogen, Carlsbad, USA)
- 5 ml of penicillin/streptomycin (final conc.: 100 U/ml/100 µg/ml; Cat 15140, Invitrogen, Carlsbad, USA)
- 10 ml of B27 supplement (Cat 17504-044, Invitrogen, Carlsbad, USA)
- 200 µM L-ascorbic acid (Cat A8960-5G, Sigma Aldrich, Taufkirchen, Germany)
- 0.22 µm sterile-filtered $CaCl_2$, titrated to 0.8 mM final concentration

EDTA solution (0.5 mM)

- 500 ml of calcium/magnesium-free PBS containing 0.45 g NaCl (Cat 14190, Invitrogen, Carlsbad, USA)
- 500 µl of 0.5 M, pH 8.0 EDTA (Cat A4892.0500, AppliChem, Darmstadt, Germany)
- Sterile-filtered (0.22 µm)

Accutase digestion medium

- Accutase® Solution (Cat SCR005, Milipore, Burlington, USA)
- 20 µg/ml DNaseI (Cat. 260913, Calbiochem, San Diego, USA)

Blocking buffer

- 500 ml of calcium/magnesium-free PBS containing 0.45 g NaCl (Cat 14190, Invitrogen, Carlsbad, USA)
- 25 ml of fetal bovine serum (FBS, Cat 10270, Invitrogen, Carlsbad, USA)
- 5 ml of bovine serum albumin (BSA, Cat A3311-50g, Sigma Aldrich, Taufkirchen, Germany)
- 2.5 ml of Triton X-100 (Cat T8787, Sigma Aldrich, Taufkirchen, Germany)

6.4 Protocols

EDTA Passaging of Wild-type and Transgenic HES-2

HES-2 were washed twice with 0.5 mM EDTA solution (room temperature), before 120 µl/cm^2 of 0.5 mM EDTA solution was added, and the cells were incubated for five minutes at room temperature. Subsequently, the EDTA was aspirated off, and the cells were rinsed off with 120 µl/cm^2 fresh E8 medium containing 10 µM ROCK inhibitor Y27632 (Cat 04-0012, Stemgent, Cambridge, USA). The cells were then plated in the culture dish of choice containing 200 µl/cm^2 fresh E8 medium with10 µM ROCK inhibitor Y27632. After 24 hours, the culture medium was changed for ROCK inhibitor-free E8 medium.

Single-cell Dissociation of HES-2-derived Cardiomyocyte Monolayers

CM were washed twice in PBS (Cat 14040, Invitrogen, Carlsbad, USA) and incubated with 0.1 ml/cm^2 Accutase® digestion medium for 10-15 minutes at 37 °C. The digestion process was stopped with threefold RPMI basal medium containing 10 µM ROCK inhibitor Y27632. Cells were then carefully triturated to single cells, centrifuged at 300 x g for five minutes, and resuspended in appropriate medium for downstream applications.

Immunolabeling of hPSC-derived Cardiomyocytes

Cells were washed twice with PBS and subsequently fixed with 4% formaldehyde in PBS (Roti® histofix 4%, Carl Roth, Karlsruhe, Germany) for 10 minutes at room temperature. Subsequently, cells were permeablized in 0.5% Triton X 100 for 10 minutes on the rocker. The dishes were washed once with PBS, before CM were blocked with 3% bovine serum albumin (BSA) for at least 30 minutes. After PBS rinsing, cells were then incubated in 0.5% Triton X for another five minutes on the rocker. The dishes were rinsed with PBS extensively, and the cells were incubated with the primary antibodies at the desired concentration in 3% BSA for at least one hour at room temperature. After washing out the primary antibodies in 0.5% Triton X for 5 minutes and extensive PBS rinsing, CM were incubated with the secondary antibodies and/or fluorophore-conjugated phalloidin for at least 30 minutes at room temperature. All incubations with antibodies was performed on a rocker. After another PBS rinsing, the remaining secondary antibodies were washed out in 0.5% Triton X for five minutes, followed by a last PBS rinsing.

PCR and Gel Electrophoresis for Genotyping of the Transgenic Cell Lines

PCR and gel electrophoresis were performed by Mrs. Tanja Gall (Drittes Physikalisches Institut, Georg-August-University Goettingen). Primers were designed by the author. 500,000 cells of each line were lysed in 50 µl of buffer, containing 50 mM KCl, 10 mM Tris pH 8.3, 2.5 mM $MgCl_2$, 0.45% Nonidet P-40, 0.45% Tween-20, 0.01% gelatin and 1 µg/ml Proteinase K at 65 °C for 30 min. Proteinase K was then inactivated by heating the samples at 95 °C for 15 min and 10 µl of sample were used as template for PCR. Primer sequences

for DNA amplification are listed in Table 6-2. The PCR products were finally separated via standard agarose gel electrophoresis.

Immunoblotting of Transgenic Cell Lines

Immunoblotting was performed by Mrs. Tanja Gall (Drittes Physikalisches Institut, Georg-August-University Goettingen). 500,000 cells were collected in 150 µl of lysis buffer, containing 150 mM NaCl, 50 mM Tris at pH 7.4, 2 mM EDTA and 1% Nonidet P-40 and were stored at -20 °C until use. For SDS-PAGE electrophoresis, 6X Laemmli loading buffer was added and the sample was heated to 95 °C for five minutes for protein denaturation. 15 µl of sample were resolved by a 12% NuPAGE Bis-Tris gel (Invitrogen) and proteins were transferred onto a polyvinylidene difluoride membrane (Immobilon-P; Millipore, Billerica, MA) using a Bio-Rad Criterion Blotter. After protein transfer, the membrane was blocked with 10% nonfat milk in PBS containing 0.05% Tween 20 for 30 minutes and incubated with affinity-purified monoclonal anti-ACTN2 antibody from mouse (A7811, Sigma-Aldrich) diluted at 1:2500 or anti-GFP antibody (11814460801; Roche) diluted at 1:500 at 4 °C overnight, followed by treatment with 1:5000 diluted peroxidase-labeled goat anti-mouse IgG (A4416, Sigma-Aldrich). The reactivity was detected with an Amersham ECL Detection Reagent (GE Healthcare Life science) using an Intas gel imager (Intas, Goettingen, Germany). The membrane probed with anti-ACTN2 antibody was treated with buffer containing 2% SDS, 100 mM β-mercaptoethanol, and 62.5 mM Tris-HCl (pH 6.8) at 50°C for 30 min to remove bound probes, and re-probed with monoclonal anti-actin antibody from mouse (clone C4; MP Biomedicals) in combination with peroxidase-labeled goat anti-mouse IgG (A4416, Sigma-Aldrich).

DNA Sequencing

Cell lysis and amplification of DNA were performed by Mrs. Tanja Gall (Drittes Physikalisches Institut, Georg-August-University Goettingen); primers were designed by the author. PCR protocols and primer sequences for DNA amplification were the same as were used for genotyping (Table 6-2). The PCR products were cloned into pCR® 2.1 vector using TA cloning® kit (Cat K2030-01, Thermo Fisher Scientific, Waltham, USA) and sequenced by Seqlab using M13 sequencing primers (Seqlab, Goettingen, Germany).

6.5 Python® Scripts

Script termed *get trajectories* to determine Z-line trajectories

```
# coding: utf-8
# In[ ]:
import sys
sys.path.append('/Users/path/functions.py')
from functions import *
get_ipython().magic(u'matplotlib notebook')
global folder
folder = None
plt.rcParams.update({'font.size': 16})

# In[ ]:
directory = '/Users/path/cellx/'
files = []
start_dir = directory
pattern   = "*.xls" # or '*.csv'
for dir,_,_ in os.walk(start_dir):
    files.extend(glob.glob(os.path.join(dir,pattern)))

print files

# In[ ]:
filename = files[0]
framerate = 18.2
folder = os.path.splitext(filename)[0]
if os.path.exists(folder)==False:
        os.mkdir(folder)
subprocess.Popen(['open',folder])
# copy image in folder
name_image = filename.split('.xls')[0] + '.png'
name_dst = folder + '/' + os.path.basename(name_image)
shutil.copyfile(name_image,name_dst)

# In[ ]:
peaks, time = get_traj_read_and_find_peaks_til(filename,folder,framerate, mph=20,
mpd=18, threshold=0, zeros_z_data=30)
np.savetxt(folder + '/peaks.txt',peaks)

# In[ ]:
peaks,time =
get_traj_read_and_find_peaks(filename,folder,framerate,delta=19,sigma=4,zeros_z_data=30)
np.savetxt(folder + '/peaks.txt',peaks)

# In[ ]:
# define frame number of rest frame
rest_frame_no = 20
np.savetxt(folder + '/rest_frame_param.txt', [rest_frame_no])
mph = 25
mpd = 18
threshold = 0
sigma = 1.8
test_x_pos,test_z_data,test_time = read_file(filename,framerate)
duarte_x_pos, duarte_z_data, duarte_time = read_file(filename,framerate)
duarte_peaks_rest = detect_peaks.detect_peaks(test_z_data[rest_frame_no,:], mph=mph,
mpd=mpd, threshold=0, edge='rising', kpsh=False, show=True)
num_of_peaks_rest = len(duarte_peaks_rest)
peaks,time =
get_traj_read_and_find_peaks_til_raw_sigma(filename,folder,framerate,num_of_peaks_rest,m
ph=mph,mpd=mpd,threshold=threshold, sigma=sigma, zeros_z_data=30)
np.savetxt(folder + '/peaks.txt',peaks)

# In[ ]:
# define frame number of rest frame
rest_frame_no = 0
np.savetxt(folder + '/rest_frame_param.txt', [rest_frame_no])
mph = 12
mpd = 17
threshold = 0
#test_x_pos,test_z_data,test_time = read_file(filename,framerate)
duarte_x_pos, duarte_z_data, duarte_time = read_file(filename,framerate)
duarte_peaks_rest = detect_peaks.detect_peaks(duarte_z_data[rest_frame_no,:], mph=mph,
mpd=mpd, threshold=threshold, edge='rising', kpsh=False, show=True)
num_of_peaks_rest = len(duarte_peaks_rest)
```

```
peaks, time = get_traj_read_and_find_peaks_til_raw(filename,folder,framerate,
num_of_peaks_rest, mph=mph, mpd=mpd, threshold=threshold, zeros_z_data=30)
np.savetxt(folder + '/peaks.txt',peaks)

# In[ ]:
# define frame number of rest frame
rest_frame_no = 0
np.savetxt(folder + '/rest_frame_param.txt', [rest_frame_no])
mph = 12
mpd = 11
threshold = 0
#test_x_pos,test_z_data,test_time = read_file(filename,framerate)
duarte_x_pos, duarte_z_data, duarte_time = read_file(filename,framerate)
duarte_peaks_rest = detect_peaks.detect_peaks(duarte_z_data[rest_frame_no,:], mph=mph,
mpd=mpd, threshold=threshold, edge='rising', kpsh=False, show=True)
num_of_peaks_rest = len(duarte_peaks_rest)
peaks, time = get_traj_read_and_find_peaks_til_raw_right(filename,folder,framerate,
num_of_peaks_rest, mph=mph, mpd=mpd, threshold=threshold, zeros_z_data=30)
np.savetxt(folder + '/peaks.txt',peaks)

# In[ ]:
pix_size_test = np.mean(np.diff(duarte_x_pos))
duarte_peaks = detect_peaks.detect_peaks(duarte_z_data[12,:], mph=12, mpd=17, thresh-
old=0, edge='rising', kpsh=False, show=True)
duarte_peaks_xpos = duarte_peaks*pix_size_test
print duarte_peaks
print duarte_peaks_xpos
print len(duarte_peaks)
peaks_1 = [pk for pk in enumerate(duarte_peaks)]
print peaks_1
np.mean(np.diff(duarte_x_pos))

# In[ ]:
z_pos_0 = get_traj_assign_peaks_1(peaks,time,folder)

# In[ ]:
z_pos_0 = get_traj_assign_peaks_1_raw(peaks,time,folder)

# In[ ]:
ref_idx = 1
z_pos,time =
get_traj_assign_peaks_2(z_pos_0,peaks,time,ref_idx,folder,dist_lims=(0.5,2.8),xlim=(0,35
))
np.savetxt(folder + '/ref_idx_param.txt', [ref_idx])
# default dist_lims=(0.5,2.8)

# In[ ]:
start_contr_float = get_x_manual(np.asarray(z_pos).T,x_figsize=60)

# In[ ]:
# save if 'False'; if already saved, then 'True' and skip step above
if False:
    start_contr = np.loadtxt(folder + '/start_contr.txt')
else:
    start_contr = np.round(start_contr_float,0)
    np.savetxt(folder + '/start_contr.txt',start_contr)

# In[ ]:
shift_contr=0
start_contr_corr =
get_corrected_start_contr_by_correlation(z_pos,framerate,start_contr,shift_contr=shift_c
ontr)
np.savetxt(folder+'/start_contr_corr.txt',start_contr_corr)
np.savetxt(folder + '/shift_contr_param.txt', [shift_contr])
# plot start times and z-line trajectories
plot_z_lines_and_start_contr(z_pos,start_contr_corr,folder)

# In[ ]:
delta_slen,equ =
get_delta_slen(z_pos,folder,start_contr_corr,framerate,plot=True,back_ind_rest=4)x

# In[ ]:
delta_slen_stereotyp =
get_stereotypical_peaks(delta_slen,framerate,start_contr_corr,folder,timeshift=0,framera
nge=1.3)
```

Script termed *analysis* to analyze parameters of contractility and inter-sarcomeric correlation

```
# coding: utf-8

# In[1]:
import sys
sys.path.append('/Users/path/functions.py')
from functions import *
get_ipython().magic(u'matplotlib notebook')
global folder
folder = None

# In[7]:
directory = '/Users/path/cellx/'
files = []
start_dir = directory
pattern   = "*.xls"
for dir,_,_ in os.walk(start_dir):
    files.extend(glob.glob(os.path.join(dir,pattern)))

# In[8]:
filename = files[0]
framerate = 18.2
folder = os.path.splitext(filename)[0]
subprocess.Popen(['open',folder])
name_of_file = filename.split('.xls')[0]
time = np.loadtxt(folder + '/time.txt')
z_pos = np.loadtxt(folder + '/z_pos.txt')
delta_slen = np.loadtxt(folder + '/delta_slen.txt')
#equ_z_pos = np.loadtxt(folder + '/equ_z_pos.txt')
equ = np.loadtxt(folder + '/equ.txt')
start_contr_corr = np.loadtxt(folder + '/start_contr_corr.txt')
delta_slen_stereotyp = np.loadtxt(folder + '/delta_slen_stereotyp.txt')

# In[9]:
idx_good = range(0,7)
# CAVE: range(0,n) is [0, 1, ..., n-1]
delta_slen_good = [delta_slen[idx] for idx in idx_good]
delta_slen_stereotyp_good = [delta_slen_stereotyp[idx] for idx in idx_good]

# In[10]:
scanrange_contr=0.9 #make larger, if time of contraction is shorter and period too
small, 0.9 good for regular 15kPa RPMI
peaks_auto,peaks_auto_diff,height_auto,vel_minus,vel_plus =
get_heights_times_period(delta_slen_good,framerate,time,start_contr_corr,scanrange_contr
,folder,figsize=(20,2))

# In[13]:
zp =
make_correlation_plots_stereotyp(delta_slen_stereotyp_good,framerate,folder,t_lim=(0,1.0
))
#zp =
make_correlation_plots_stereotyp(delta_slen_stereotyp_good,framerate,folder,t_lim=(0,0.7
1))

# In[14]:
make_correlation_plots(delta_slen_good,framerate,start_contr_corr,folder,t_lim=(0,1.0))
make_correlation_plot_single_sarcomer(delta_slen_good,framerate,start_contr_corr,folder)

# In[16]:
idx_plot = np.arange(0,7,1)
#idx_plot = np.arange(4,12,1)
yticks = [-0.2,-0.1,0,0.2]
make_neighbor_correlation_plot(delta_slen_stereotyp,framerate,folder,idx_plot,yticks=yti
cks,xrange_neighbor=1.3)

# In[17]:
corr_stereotype_loaded = np.load(folder + '/correlation/corr_dist_time_stereotyp.npy')
corr_all_loaded = np.load(folder + '/correlation/corr_dist_time.npy')
corr_stereotype_vel_loaded = np.load(folder +
'/correlation/corr_dist_time_vel_stereotyp.npy')
#corr_all_vel_loaded = np.load(folder + '/correlation/corr_dist_time_vel.npy')
# define time interval of maximal contraction correlation
time_int_of_max_contract = np.argmax(corr_stereotype_loaded,0)[0]
#time_int_of_max_contract = 15
time_int_of_all_max_contract = np.argmax(corr_all_loaded,0)[0]
```

```
#time_int_of_all_max_contract = 15
# define time intervals of maximal velocity correlation (contract and relax)
time_int_of_max_vel_contract =
np.argmax(corr_stereotype_vel_loaded[:time_int_of_max_contract],0)[0]
#time_int_of_max_vel_contract = 9
time_int_of_max_vel_relax = time_int_of_max_contract +
np.argmax(corr_stereotype_vel_loaded[time_int_of_max_contract:],0)[0]
#time_int_of_max_vel_relax = 12
corr_stereo_at_max_contract = corr_stereotype_loaded[time_int_of_max_contract][:]
corr_stereo_at_max_contract_norm =
corr_stereo_at_max_contract/corr_stereotype_loaded[time_int_of_max_contract][0]
corr_all_at_max_contract = corr_all_loaded[time_int_of_all_max_contract][:]
corr_all_at_max_contract_norm =
corr_all_at_max_contract/corr_all_loaded[time_int_of_all_max_contract][0]
corr_stereo_at_max_vel_contract =
corr_stereotype_vel_loaded[time_int_of_max_vel_contract][:]
corr_stereo_at_max_vel_contract_norm =
corr_stereo_at_max_vel_contract/corr_stereotype_vel_loaded[time_int_of_max_vel_contract]
[0]
corr_stereo_at_max_vel_relax = corr_stereotype_vel_loaded[time_int_of_max_vel_relax][:]
corr_stereo_at_max_vel_relax_norm =
corr_stereo_at_max_vel_relax/corr_stereotype_vel_loaded[time_int_of_max_vel_relax][0]
np.savetxt(folder + '/correlation/corr_stereo_at_max_contract.txt',
corr_stereo_at_max_contract)
np.savetxt(folder + '/summary_correlation_contraction.txt',
corr_stereo_at_max_contract_norm, fmt='%.5g')
np.savetxt(folder + '/summary_correlation_velocity_contr.txt',
corr_stereo_at_max_vel_contract_norm, fmt='%.5g')
np.savetxt(folder + '/summary_correlation_velocity_relax.txt',
corr_stereo_at_max_vel_relax_norm, fmt='%.5g')
np.savetxt(folder + '/summary_correlation.txt', corr_stereo_at_max_contract_norm,
fmt='%.5g')
f = open(folder + '/summary_correlation.txt', 'a')
np.savetxt(f,corr_all_at_max_contract_norm, fmt='%.5g')
np.savetxt(f,corr_stereo_at_max_vel_contract_norm, fmt='%.5g')
np.savetxt(f,corr_stereo_at_max_vel_relax_norm, fmt='%.5g')
f.close()
np.savetxt(folder + '/correlation/corr_all_at_max_contract.txt',
corr_all_at_max_contract)
print ('Time interval of stereotype maximal contraction correlation is: %s' %
(time_int_of_max_contract))
print ('corresponding to [s]: %s' % (time_int_of_max_contract/framerate))
print corr_stereo_at_max_contract_norm
print ('Time interval of all maximal contraction correaltion is: %s' %
(time_int_of_all_max_contract))
print corr_all_at_max_contract_norm
print corr_stereo_at_max_vel_contract_norm
print corr_stereo_at_max_vel_relax_norm
#print time_int_of_all_max_contract

# In[18]:
period_to_save = float(np.loadtxt(folder + '/period.txt')) #period_script
mean_period_to_save = np.loadtxt(folder + '/mean_diff-std_diff-mean_freq-
std_freq.txt')[0]
std_period_to_save = np.loadtxt(folder + '/mean_diff-std_diff-mean_freq-
std_freq.txt')[1]
mean_freq_to_save = np.loadtxt(folder + '/mean_diff-std_diff-mean_freq-std_freq.txt')[2]
std_freq_to_save = np.loadtxt(folder + '/mean_diff-std_diff-mean_freq-std_freq.txt')[3]
rest_slen_from_restframe = np.loadtxt(folder + '/rest_slen.txt')
mean_rest_slen = np.nanmean(rest_slen_from_restframe)
median_rest_slen = np.nanmedian(rest_slen_from_restframe)
std_rest_slen = np.nanstd(rest_slen_from_restframe)
zero_slen = np.loadtxt(folder + '/equ.txt')
mean_zero_slen = np.nanmean(zero_slen)
median_zero_slen = np.nanmedian(zero_slen)
std_zero_slen = np.nanstd(zero_slen)
avg_maxcontraction = abs(np.nanmean(height_auto))
median_maxcontraction = abs(np.nanmedian(height_auto))
std_maxcontraction = np.nanstd(height_auto, ddof=1)
max_maxcontraction = abs(np.amin(height_auto))
min_maxcontraction = abs(np.amax(height_auto))
avg_velplus = np.nanmean(vel_plus*framerate)
std_velplus = np.nanstd(vel_plus*framerate, ddof=1)
avg_velminus = np.nanmean(vel_minus*framerate)
std_velminus = np.nanstd(vel_minus*framerate, ddof=1)
avg_time_of_maxcontraction = np.nanmean(peaks_auto_diff/framerate)
median_time_of_maxcontraction = np.nanmedian(peaks_auto_diff/framerate)
std_time_of_maxcontraction = np.nanstd(peaks_auto_diff/framerate, ddof=1)
```

```
num_sarcomeres_analyzed = len(height_auto)
num_beatings_analyzed = len(height_auto[0])
vector_to_save_output = [period_to_save, mean_period_to_save, std_period_to_save,
mean_freq_to_save, std_freq_to_save,
                         mean_rest_slen, median_rest_slen, std_rest_slen,
mean_zero_slen, median_zero_slen, std_zero_slen,
                  avg_maxcontraction, median_maxcontraction, std_maxcontraction,
max_maxcontraction, min_maxcontraction,
                  avg_velplus, std_velplus, avg_velminus, std_velminus,
avg_time_of_maxcontraction, median_time_of_maxcontraction, std_time_of_maxcontraction,
                  num_sarcomeres_analyzed, num_beatings_analyzed]
np.savetxt(folder + '/summary_vector_output.txt', np.atleast_2d(vector_to_save_output),
fmt='%.5g', delimiter="\t")

# In[19]:
mph_param = np.loadtxt(folder + '/mph_mpd_thresh_param.txt')[0]
mpd_param = np.loadtxt(folder + '/mph_mpd_thresh_param.txt')[1]
threshold_param = np.loadtxt(folder + '/mph_mpd_thresh_param.txt')[2]
dist_min_param = np.loadtxt(folder + '/dist_lims_param.txt')[0]
dist_max_param = np.loadtxt(folder + '/dist_lims_param.txt')[1]
zeros_z_data_param = float(np.loadtxt(folder + '/zeros_z_data_param.txt'))
ref_idx_param = float(np.loadtxt(folder + '/ref_idx_param.txt'))
shift_contr_param = float(np.loadtxt(folder + '/shift_contr_param.txt'))
back_ind_param = float(np.loadtxt(folder + '/back_ind_rest_param.txt'))
timeshift_param = np.loadtxt(folder + '/timeshift_framerange_param.txt')[0]
framerange_param = np.loadtxt(folder + '/timeshift_framerange_param.txt')[1]
rest_frame_param = float(np.loadtxt(folder + '/rest_frame_param.txt'))
vector_to_save_input = [framer-
ate,mph_param,mpd_param,threshold_param,dist_min_param,dist_max_param,zeros_z_data_param
,ref_idx_param,

shift_contr_param,back_ind_param,timeshift_param,framerange_param,rest_frame_param,

time_int_of_max_contract,time_int_of_all_max_contract,time_int_of_max_vel_contract,time_
int_of_max_vel_relax,scanrange_contr]
np.savetxt(folder + '/summary_vector_input.txt', np.atleast_2d(vector_to_save_input),
fmt='%.5g', delimiter="\t")
```

Functions.py, which is called in the two above scripts

```
import numpy as np
import h5py
import matplotlib.pyplot as plt
import peakutils as pu
from scipy import signal
import peakdetect
import detect_peaks
#import matplotlib.cm as cmx
#import matplotlib.colors as colors
import glob
import os
import pylab as plb
#from statistics import mode
import subprocess
import scipy
from mpl_toolkits.mplot3d import Axes3D
from matplotlib import colors as mcolors
from shutil import copyfile
import shutil
global folder
def read_file(filename,framerate):
#read data
#data = pd.read_csv(filename, delimiter='\t').T
    file_format = os.path.splitext(filename)[1]
    if file_format=='.csv':
        data = np.genfromtxt(filename,delimiter=',').T
        y_int = data.T[1:].T[2:]
        x_pos = data[1][1:]
if file_format=='.xls':
        data = np.genfromtxt(filename,delimiter='\t').T
        y_int = data.T[1:].T[2:]
        x_pos = data[1][1:]
#framerate -> time
    no_frames = len(y_int)
    print no_frames
    framerate = float(framerate)
    time = np.arange(0,no_frames/framerate,1/framerate)
```

```
    return x_pos,y_int,time
def detekt_peaks(x_pos,y_int,delta=40,sigma=5,zeros_z_data=30,plot=False):
#def de-
tekt_peaks(x_pos,y_int,thres=0.3,min_dist=10,delta=40,sigma=5,zeros_z_data=30,plot=False
):
    z_data = np.zeros((len(y_int),zeros_z_data))
    #z_data = np.zeros((len(y_int),30))
    #print delta, sigma, zeros_z_data
#print len(y_int)
i= 0
    for y in y_int:
        #idx_1 = pu.indexes(y,thres=thres,min_dist=min_dist)
        #idx_0 = signal.find_peaks_cwt(y,widths=np.arange(20,300))
        #print peakdetect.peakdet(y,delta=10,x=x_pos)
        #peaks = peakdetect.peakdet(y,delta=5,x=x_pos)[0]
        #peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)[0].T[0]
        peaks_0 = peakdetekt_MATLAB(x_pos,y,delta=delta,sigma=sigma)
        #peak_pos = x_pos[idx_1]
        #peak_height = y[idx_1]
        #try:
        #    peak_pos = pu.interpolate(x_pos,y,ind=idx_1,width=10)
            #peak_pos = x_pos[idx_0]
            #print "correct"
        #except:
        #    pass
            #idx_1 = [int(x) for x in idx_1]
            #peak_pos = x_pos[idx_1]
            #print "else"
        #print peak_pos
        z_data[i][:len(peaks_0)] = peaks_0
        z_data[i][z_data[i]==0] = np.nan
        if plot==True:
            plt.figure()
            plt.plot(x_pos,y_int[i],c='b')
            for peak_pos in peaks_0:
plt.plot([peak_pos,peak_pos],[np.nanmin(y_int[i]),np.nanmax(y_int[i])],marker='x',color=
'r',s=50)
            plt.ylabel('Intensity')
            plt.xlabel('x in um')
            plt.show()
        i+=1
    print z_data.shape
    return z_data.T
def detekt_peaks_til_raw_sigma(x_pos,y_int, pix_size, num_of_peaks_rest, mph=1, mpd=12,
threshold=0.01,sigma=5,zeros_z_data=30,plot=False):
#def de-
tekt_peaks(x_pos,y_int,thres=0.3,min_dist=10,delta=40,sigma=5,zeros_z_data=30,plot=False
):
    z_data = np.zeros((len(y_int),zeros_z_data))
    #z_data = np.zeros((len(y_int),30))
    #print delta, sigma, zeros_z_data
    #print len(y_int)
    i= 0
    for y in y_int:
        #idx_1 = pu.indexes(y,thres=thres,min_dist=min_dist)
        #idx_0 = signal.find_peaks_cwt(y,widths=np.arange(20,300))
        #print peakdetect.peakdet(y,delta=10,x=x_pos)
        #peaks = peakdetect.peakdet(y,delta=5,x=x_pos)[0]
        #peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)[0].T[0]
        peaks_0 = peakdetekt_MATLAB_raw_sigma(x_pos,y, pix_size,num_of_peaks_rest,
mph=mph, mpd=mpd, threshold=threshold,sigma=sigma)
        #peak_pos = x_pos[idx_1]
        #peak_height = y[idx_1]
        #try:
        #    peak_pos = pu.interpolate(x_pos,y,ind=idx_1,width=10)
            #peak_pos = x_pos[idx_0]
            #print "correct"
        #except:
        #    pass
            #idx_1 = [int(x) for x in idx_1]
            #peak_pos = x_pos[idx_1]
            #print "else"
        #print peak_pos
        z_data[i][:len(peaks_0)] = peaks_0
        z_data[i][z_data[i]==0] = np.nan
        if plot==True:
            plt.figure()
            plt.plot(x_pos,y_int[i],c='b')
```

```
            for peak_pos in peaks_0:
plt.plot([peak_pos,peak_pos],[np.nanmin(y_int[i]),np.nanmax(y_int[i])],marker='x',color=
'r',s=50)
            plt.ylabel('Intensity')
            plt.xlabel('x in um')
            plt.show()
        i+=1
    print z_data.shape
    return z_data.T
def detekt_peaks_til(x_pos, y_int, pix_size, mph=1, mpd=12, threshold=0.01, ze-
ros_z_data=30, plot=False):
#def detekt_peaks_til(x_pos, y_int, pix_size, mpd=12, threshold=0.01, delta=40, sigma=5,
zeros_z_data=30, plot=False):
    z_data = np.zeros((len(y_int),zeros_z_data))
    #z_data = np.zeros((len(y_int),30))
    #print delta, sigma, zeros_z_data
    #print len(y_int)
    i= 0
    for y in y_int:
        #idx_1 = pu.indexes(y,thres=thres,min_dist=min_dist)
        #idx_0 = signal.find_peaks_cwt(y,widths=np.arange(20,300))
        #print peakdetect.peakdet(y,delta=10,x=x_pos)
        #peaks = peakdetect.peakdet(y,delta=5,x=x_pos)[0]
        #peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)[0].T[0]
        peaks_0 = peakdetekt_MATLAB_til(x_pos, y, pix_size, mph=mph, mpd=mpd, thresh-
old=threshold)
        #peaks_0 = peakdetekt_MATLAB_til(x_pos, y, pix_size, mpd=mpd, thresh-
old=threshold, delta=delta,sigma=sigma)
        #peak_pos = x_pos[idx_1]
        #peak_height = y[idx_1]
        #try:
        #    peak_pos = pu.interpolate(x_pos,y,ind=idx_1,width=10)
            #peak_pos = x_pos[idx_0]
            #print "correct"
        #except:
        #    pass
            #idx_1 = [int(x) for x in idx_1]
            #peak_pos = x_pos[idx_1]
            #print "else"
        #print peak_pos
        z_data[i][:len(peaks_0)] = peaks_0
        z_data[i][z_data[i]==0] = np.nan
        if plot==True:
            plt.figure()
            plt.plot(x_pos,y_int[i],c='b')
            for peak_pos in peaks_0:
plt.plot([peak_pos,peak_pos],[np.nanmin(y_int[i]),np.nanmax(y_int[i])],marker='x',color=
'r',s=50)
            plt.ylabel('Intensity')
            plt.xlabel('x in um')
            plt.show()
        i+=1
    print z_data.shape
    return z_data.T
def detekt_peaks_til_raw(x_pos, y_int, pix_size, num_of_peaks_rest, mph=1, mpd=12,
threshold=0.01, zeros_z_data=30, plot=False):
#def detekt_peaks_til(x_pos, y_int, pix_size, mpd=12, threshold=0.01, delta=40, sigma=5,
zeros_z_data=30, plot=False):
    z_data = np.zeros((len(y_int),zeros_z_data))
    #z_data = np.zeros((len(y_int),30))
    #print delta, sigma, zeros_z_data
    #print len(y_int)
    i= 0
    for y in y_int:
        #idx_1 = pu.indexes(y,thres=thres,min_dist=min_dist)
        #idx_0 = signal.find_peaks_cwt(y,widths=np.arange(20,300))
        #print peakdetect.peakdet(y,delta=10,x=x_pos)
        #peaks = peakdetect.peakdet(y,delta=5,x=x_pos)[0]
        #peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)[0].T[0]
        peaks_0 = peakdetekt_MATLAB_til_raw(x_pos, y, pix_size, num_of_peaks_rest,
mph=mph, mpd=mpd, threshold=threshold)
        #peaks_0 = peakdetekt_MATLAB_til(x_pos, y, pix_size, mpd=mpd, thresh-
old=threshold, delta=delta,sigma=sigma)
        #peak_pos = x_pos[idx_1]
        #peak_height = y[idx_1]
        #try:
        #    peak_pos = pu.interpolate(x_pos,y,ind=idx_1,width=10)
            #peak_pos = x_pos[idx_0]
```

```
            #print "correct"
        #except:
        #    pass
            #idx_1 = [int(x) for x in idx_1]
            #peak_pos = x_pos[idx_1]
            #print "else"
        #print peak_pos
        z_data[i][:len(peaks_0)] = peaks_0
        z_data[i][z_data[i]==0] = np.nan
        if plot==True:
            plt.figure()
            plt.plot(x_pos,y_int[i],c='b')
            for peak_pos in peaks_0:
plt.plot([peak_pos,peak_pos],[np.nanmin(y_int[i]),np.nanmax(y_int[i])],marker='x',color=
'r',s=50)
            plt.ylabel('Intensity')
            plt.xlabel('x in um')
            plt.show()
        i+=1
    print z_data.shape
    return z_data.T
def detekt_peaks_til_raw_right(x_pos, y_int, pix_size, num_of_peaks_rest, mph=1, mpd=12,
threshold=0.01, zeros_z_data=30, plot=False):
#def detekt_peaks_til(x_pos, y_int, pix_size, mpd=12, threshold=0.01, delta=40, sigma=5,
zeros_z_data=30, plot=False):
    z_data = np.zeros((len(y_int),zeros_z_data))
    #z_data = np.zeros((len(y_int),30))
    #print delta, sigma, zeros_z_data
    #print len(y_int)
    i= 0
    for y in y_int:
        #idx_1 = pu.indexes(y,thres=thres,min_dist=min_dist)
        #idx_0 = signal.find_peaks_cwt(y,widths=np.arange(20,300))
        #print peakdetect.peakdet(y,delta=10,x=x_pos)
        #peaks = peakdetect.peakdet(y,delta=5,x=x_pos)[0]
        #peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)[0].T[0]
        peaks_0 = peakdetekt_MATLAB_til_raw_right(x_pos, y, pix_size, num_of_peaks_rest,
mph=mph, mpd=mpd, threshold=threshold)
        #peaks_0 = peakdetekt_MATLAB_til(x_pos, y, pix_size, mpd=mpd, thresh-
old=threshold, delta=delta,sigma=sigma)
        #peak_pos = x_pos[idx_1]
        #peak_height = y[idx_1]
        #try:
        #    peak_pos = pu.interpolate(x_pos,y,ind=idx_1,width=10)
            #peak_pos = x_pos[idx_0]
            #print "correct"
        #except:
        #    pass
            #idx_1 = [int(x) for x in idx_1]
            #peak_pos = x_pos[idx_1]
            #print "else"
        #print peak_pos
        z_data[i][:len(peaks_0)] = peaks_0
        z_data[i][z_data[i]==0] = np.nan
        if plot==True:
            plt.figure()
            plt.plot(x_pos,y_int[i],c='b')
            for peak_pos in peaks_0:
plt.plot([peak_pos,peak_pos],[np.nanmin(y_int[i]),np.nanmax(y_int[i])],marker='x',color=
'r',s=50)
            plt.ylabel('Intensity')
            plt.xlabel('x in um')
            plt.show()
        i+=1
    print z_data.shape
    return z_data.T
def peakdetekt_MATLAB(x_pos,y,delta=40,sigma=4):
    peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)
    peaks = peaks_0[0].T[0]
    height = peaks_0[0].T[1]
    height_lim = np.nanmean(height)-sigma*np.nanstd(height)
    peaks_1 = [pk for i,pk in enumerate(peaks) if height[i]>height_lim]
    #np.savetxt(folder + '/peaks_matlab.txt',peaks_0[0].T[0])
    return peaks_1
def peakdetekt_MATLAB_raw_sigma(x_pos,y,pix_size, num_of_peaks_rest, mph=1, mpd=12,
threshold=0.01,sigma=4):
    #peaks_0 = peakdetect.peakdet(y,delta=delta,x=x_pos)
    peaks = detect_peaks.detect_peaks(y, mph=mph, mpd=mpd, threshold=threshold)
```

```
    #peaks = peaks_0[0].T[0]
    height = [y[k] for m,k in enumerate(peaks)]
    height_lim = np.nanmean(height)-sigma*np.nanstd(height)
    #height_lim = 0
    peaks_1 = np.asarray([pk for i,pk in enumerate(peaks) if height[i]>height_lim])
    duarte_peaks=[]
    if len(peaks_1) == num_of_peaks_rest:
         duarte_peaks = peaks_1*pix_size
    #if len(peaks) > num_of_peaks_rest:
    else:
         duarte_peaks = peaks_1[1:]*pix_size
    #print duarte_peaks
    peaks_2=duarte_peaks
    #np.savetxt(folder + '/peaks_matlab.txt',peaks_0[0].T[0])
    return peaks_2
def peakdetekt_MATLAB_til(x_pos, y, pix_size, mph=1, mpd=12, threshold=0.01):
#def peakdetekt_MATLAB_til(x_pos, y, pix_size, mpd=12, threshold=0.01, delta=40, sig-
ma=4):
    #peaks_0 = detect_peaks.detect_peaks(y, mpd=mpd, threshold=threshold)
    #peaks = peaks_0[0].T[0]
    #height = peaks_0[0].T[1]
    #height_lim = np.nanmean(height)-sigma*np.nanstd(height)
    #peaks_1 = [pk for i,pk in enumerate(peaks) if height[i]>height_lim]
    peaks = detect_peaks.detect_peaks(y, mph=mph, mpd=mpd, threshold=threshold)
    #peaks_1 = [pk for pk in enumerate(peaks)]
    peaks_1 = peaks*pix_size
    #np.savetxt(folder + '/peaks_matlab.txt',peaks_0[0].T[0])
    return peaks_1
def peakdetekt_MATLAB_til_raw(x_pos, y, pix_size, num_of_peaks_rest, mph=1, mpd=12,
threshold=0.01):
#def peakdetekt_MATLAB_til(x_pos, y, pix_size, mpd=12, threshold=0.01, delta=40, sig-
ma=4):
    #peaks_0 = detect_peaks.detect_peaks(y, mpd=mpd, threshold=threshold)

    #peaks = peaks_0[0].T[0]
    #height = peaks_0[0].T[1]
    #height_lim = np.nanmean(height)-sigma*np.nanstd(height)
    #peaks_1 = [pk for i,pk in enumerate(peaks) if height[i]>height_lim]
    peaks = detect_peaks.detect_peaks(y, mph=mph, mpd=mpd, threshold=threshold)
    #peaks_1 = [pk for pk in enumerate(peaks)]
    duarte_peaks=[]
    if len(peaks) == num_of_peaks_rest:
         duarte_peaks = peaks*pix_size
    #if len(peaks) > num_of_peaks_rest:
    else:
         duarte_peaks = peaks[1:]*pix_size
    peaks_1 = duarte_peaks
    #np.savetxt(folder + '/peaks_matlab.txt',peaks_0[0].T[0])
    return peaks_1
def peakdetekt_MATLAB_til_raw_right(x_pos, y, pix_size, num_of_peaks_rest, mph=1,
mpd=12, threshold=0.01):
#def peakdetekt_MATLAB_til(x_pos, y, pix_size, mpd=12, threshold=0.01, delta=40, sig-
ma=4):
    #peaks_0 = detect_peaks.detect_peaks(y, mpd=mpd, threshold=threshold)

    #peaks = peaks_0[0].T[0]
    #height = peaks_0[0].T[1]

    #height_lim = np.nanmean(height)-sigma*np.nanstd(height)
    #peaks_1 = [pk for i,pk in enumerate(peaks) if height[i]>height_lim]
    peaks = detect_peaks.detect_peaks(y, mph=mph, mpd=mpd, threshold=threshold)
    #peaks_1 = [pk for pk in enumerate(peaks)]
    duarte_peaks=[]
    if len(peaks) == num_of_peaks_rest+2:
         duarte_peaks = peaks[1:]*pix_size
         #duarte_peaks = peaks[1:num_of_peaks_rest-1]*pix_size #use only when very large
displacement
    #if len(peaks) > num_of_peaks_rest:
    else:
         duarte_peaks = peaks*pix_size
         #duarte_peaks = peaks[:num_of_peaks_rest-1]*pix_size #use only when very large
displacement
    peaks_1 = duarte_peaks
    #np.savetxt(folder + '/peaks_matlab.txt',peaks_0[0].T[0])
    return peaks_1
def get_traj_read_and_find_peaks(filename,folder,framerate,delta=40,sigma=5, ze-
ros_z_data=30):
    #tils test: x_pos,y_int,time = read_file(filename,framerate)
```

```
    x_pos,z_data,time = read_file(filename,framerate)
    #tils test: peaks = detekt_peaks(x_pos,y_int,delta=delta,sigma=sigma, ze-
ros_z_data=zeros_z_data)
    peaks = detekt_peaks(x_pos,z_data,delta=delta,sigma=sigma, ze-
ros_z_data=zeros_z_data)
    delta_sigma_crit = [delta, sigma]
    np.savetxt(folder + '/delta_sigma_param.txt', delta_sigma_crit)
    zeros_z_data_param = [zeros_z_data]
    np.savetxt(folder + '/zeros_z_data_param.txt', zeros_z_data_param)
    return peaks,time
def get_traj_read_and_find_peaks_til(filename,folder,framerate,mph=1, mpd=12, thresh-
old=0.01, zeros_z_data=30):
#def get_traj_read_and_find_peaks_til(filename,folder,framerate,mpd=12, thresh-
old=0.01,delta=40,sigma=5, zeros_z_data=30):
    x_pos,y_int,time = read_file(filename,framerate)
    pix_size = np.mean(np.diff(x_pos))
    peaks = detekt_peaks_til(x_pos, y_int, pix_size, mph=mph, mpd=mpd, thresh-
old=threshold, zeros_z_data=zeros_z_data)
    #peaks = detekt_peaks_til(x_pos, y_int, pix_size, mpd=mpd, thresh-
old=threshold,delta=delta,sigma=sigma,zeros_z_data=zeros_z_data)
    #delta_sigma_crit = [delta, sigma]
    mpd_thresh_crit = [mph, mpd, threshold]
    np.savetxt(folder + '/mph_mpd_thresh_param.txt', mpd_thresh_crit)
    #np.savetxt(folder + '/delta_sigma_param.txt', delta_sigma_crit)
    zeros_z_data_param = [zeros_z_data]
    np.savetxt(folder + '/zeros_z_data_param.txt', zeros_z_data_param)
    return peaks,time
def
get_traj_read_and_find_peaks_til_raw(filename,folder,framerate,num_of_peaks_rest,mph=1,
mpd=12, threshold=0.01, zeros_z_data=30):
#def get_traj_read_and_find_peaks_til(filename,folder,framerate,mpd=12, thresh-
old=0.01,delta=40,sigma=5, zeros_z_data=30):
    x_pos,y_int,time = read_file(filename,framerate)
    pix_size = np.mean(np.diff(x_pos))
    peaks = detekt_peaks_til_raw(x_pos, y_int, pix_size, num_of_peaks_rest, mph=mph,
mpd=mpd, threshold=threshold, zeros_z_data=zeros_z_data)
    #peaks = detekt_peaks_til(x_pos, y_int, pix_size, mpd=mpd, thresh-
old=threshold,delta=delta,sigma=sigma,zeros_z_data=zeros_z_data)
    #delta_sigma_crit = [delta, sigma]
    mpd_thresh_crit = [mph, mpd, threshold]
    np.savetxt(folder + '/mph_mpd_thresh_param.txt', mpd_thresh_crit)
    #np.savetxt(folder + '/delta_sigma_param.txt', delta_sigma_crit)
    zeros_z_data_param = [zeros_z_data]
    np.savetxt(folder + '/zeros_z_data_param.txt', zeros_z_data_param)
    return peaks,time
def
get_traj_read_and_find_peaks_til_raw_right(filename,folder,framerate,num_of_peaks_rest,m
ph=1, mpd=12, threshold=0.01, zeros_z_data=30):
#def get_traj_read_and_find_peaks_til(filename,folder,framerate,mpd=12, thresh-
old=0.01,delta=40,sigma=5, zeros_z_data=30):
    x_pos,y_int,time = read_file(filename,framerate)
    pix_size = np.mean(np.diff(x_pos))
    peaks = detekt_peaks_til_raw_right(x_pos, y_int, pix_size, num_of_peaks_rest,
mph=mph, mpd=mpd, threshold=threshold, zeros_z_data=zeros_z_data)
    #peaks = detekt_peaks_til(x_pos, y_int, pix_size, mpd=mpd, thresh-
old=threshold,delta=delta,sigma=sigma,zeros_z_data=zeros_z_data)
    #delta_sigma_crit = [delta, sigma]
    mpd_thresh_crit = [mph, mpd, threshold]
    np.savetxt(folder + '/mph_mpd_thresh_param.txt', mpd_thresh_crit)
    #np.savetxt(folder + '/delta_sigma_param.txt', delta_sigma_crit)
    zeros_z_data_param = [zeros_z_data]
    np.savetxt(folder + '/zeros_z_data_param.txt', zeros_z_data_param)
    return peaks,time
def
get_traj_read_and_find_peaks_til_raw_sigma(filename,folder,framerate,num_of_peaks_rest,m
ph=1,mpd=12,threshold=0, sigma=5, zeros_z_data=30):
#def get_traj_read_and_find_peaks_til(filename,folder,framerate,mpd=12, thresh-
old=0.01,delta=40,sigma=5, zeros_z_data=30):
    x_pos,y_int,time = read_file(filename,framerate)
    pix_size = np.mean(np.diff(x_pos))
    peaks = detekt_peaks_til_raw_sigma(x_pos, y_int, pix_size,num_of_peaks_rest,
mph=mph, mpd=mpd, threshold=threshold, sigma=sigma, zeros_z_data=zeros_z_data)
    #peaks = detekt_peaks_til(x_pos, y_int, pix_size, mpd=mpd, thresh-
old=threshold,delta=delta,sigma=sigma,zeros_z_data=zeros_z_data)
    #delta_sigma_crit = [delta, sigma]
    mpd_thresh_crit = [mph, mpd, sigma]
    np.savetxt(folder + '/mph_mpd_thresh_param.txt', mpd_thresh_crit)
    #np.savetxt(folder + '/delta_sigma_param.txt', delta_sigma_crit)
```

```
    zeros_z_data_param = [zeros_z_data]
    np.savetxt(folder + '/zeros_z_data_param.txt', zeros_z_data_param)
    return peaks,time
def get_traj_assign_peaks_1(peaks,time,folder):
    start_pos = [pos for pos in peaks.T[0] if np.isnan(pos)==False]
    z_pos = np.zeros((len(start_pos),len(time)))*np.nan
    for i,z_line_start in enumerate(start_pos):
        z_pos[i,0]= z_line_start
        for j,z_pos_j in enumerate(z_pos[i]):
            if j>0 and j<len(peaks.T):
                curr_pos = z_pos[i,j-1]
                next_peaks = peaks.T[j]
                dist_next = np.abs([peak-curr_pos for peak in next_peaks])
                #print dist_next
                next_idx = np.nanargmin(dist_next)
                #print next_idx
                z_pos[i,j] = next_peaks[next_idx]
                curr_pos = z_pos[i,j]
    plt.figure(figsize=(10,5))
    plt.plot(time,z_pos.T)
    plt.xlabel('time [s]')
    plt.ylabel('z-line position [um]')
    plt.savefig(folder + '/z_pos_0.png',format='png')
    np.savetxt(folder + '/z_pos_0.txt',z_pos)
    return z_pos
def get_traj_assign_peaks_1_raw(peaks,time,folder):
    start_pos = [pos for pos in peaks.T[0] if np.isnan(pos)==False]
    z_pos = np.zeros((len(start_pos),len(time)))*np.nan
    for i,z_line_start in enumerate(start_pos):
        z_pos[i,0]= z_line_start
        for j,z_pos_j in enumerate(z_pos[i]):
            if j>0 and j<len(peaks.T):
                curr_pos = z_pos[i,j-1]
                next_peaks = peaks.T[j]
                #dist_next = np.abs([peak-curr_pos for peak in next_peaks])
                #print dist_next
                #next_idx = np.nanargmin(dist_next)
                next_idx = i
                #print next_idx
                z_pos[i,j] = next_peaks[next_idx]
                curr_pos = z_pos[i,j]
    plt.figure(figsize=(10,5))
    plt.plot(time,z_pos.T)
    plt.xlabel('time [s]')
    plt.ylabel('z-line position [um]')
    plt.savefig(folder + '/z_pos_0.png',format='png')
    return z_pos
def
get_traj_assign_peaks_2(z_pos,peaks,time,ref_idx,folder,dist_lims=(1,2.4),xlim=None):
    z_pos_corr = z_pos.copy()
    list_plus  = range(ref_idx,len(z_pos))
    list_minus = list(np.arange(ref_idx,-1,-1))
    for i,idx in enumerate(list_plus):
        if i>0:
            z_i = z_pos[idx]
            z_i_corr = z_i.copy()
            ref_z = z_pos_corr[list_plus[i-1]]
            for j,z_j in enumerate(z_i[:-1]):
                diff = z_i[j] - ref_z[j]
                peaks_j = peaks.T[j]
                ref_idx = np.nanargmin(np.abs([peak-ref_z[j] for peak in peaks_j]))
                peak_idx = np.nanargmin(np.abs([peak-z_j for peak in peaks_j]))
                if diff> dist_lims[1]:
                    z_i_corr[j] = peaks_j[peak_idx-1]
                if diff<dist_lims[0]:
                    z_i_corr[j] = peaks_j[peak_idx+1]
                if z_i_corr[j]== ref_z[j]:
                    z_i_corr[j] = peaks_j[peak_idx]
                #if (peak_idx-ref_idx)>1 or peak_idx-ref_idx<0:
                    #z_i_corr[j] = peaks_j[ref_idx+1]
            z_pos_corr[idx] = z_i_corr
    for i,idx in enumerate(list_minus):
        if i>0:
            z_i = z_pos[idx]
            z_i_corr = z_i.copy()
            ref_z = z_pos_corr[list_minus[i-1]]
            for j,z_j in enumerate(z_i[:-1]):
                diff = z_i[j] - ref_z[j]
```

```
                peaks_j = peaks.T[j]
                ref_idx = np.nanargmin(np.abs([peak-ref_z[j] for peak in peaks_j]))
                peak_idx = np.nanargmin(np.abs([peak-z_j for peak in peaks_j]))
                if diff<- dist_lims[1]:
                    z_i_corr[j] = peaks_j[peak_idx+1]
                if diff> -dist_lims[0]:
                    z_i_corr[j] = peaks_j[peak_idx-1]
                if z_i_corr[j]== ref_z[j]:
                    z_i_corr[j] = peaks_j[peak_idx]
                #if (ref_idx-peak_idx)>1 or (ref_idx-peak_idx)<0:
                   # z_i_corr[j] = peaks_j[ref_idx-1]
            z_pos_corr[idx] = z_i_corr
    show_peaks = False
    fig = plt.figure(figsize=(10,5))
    fig.add_subplot(1,1,1)
    if xlim<>None:
        xlim_idx = [i for i,t in enumerate(time) if t>=xlim[0] and t<xlim[1]]
        time_xlim = time[xlim_idx]
        z_pos_xlim = z_pos_corr.T[xlim_idx].T
        plt.plot(time_xlim,z_pos_xlim.T,'k')
    else:
        plt.plot(time,z_pos_corr.T)
    if show_peaks:
        for peak in peaks:
            plt.scatter(time,peak,marker='x',s=10)
    plt.xlim((0,max(time)))
    plt.xlabel('time [s]')
    plt.ylabel('z-line position [um]')
    plt.savefig(folder + '/z_pos.png',format='png')
    np.savetxt(folder + '/peaks_assign2.txt',peaks)
    dist_lims_param = [dist_lims[0], dist_lims[1]]
    np.savetxt(folder + '/dist_lims_param.txt', dist_lims_param)
    if xlim<>None:
        np.savetxt(folder + '/z_pos.txt',z_pos_xlim)
        np.savetxt(folder + '/time.txt',time_xlim)
        return z_pos_xlim,time_xlim
    else:
        np.savetxt(folder + '/z_pos.txt',z_pos_corr)
        return z_pos_corr,time
def
plot_z_lines_and_start_contr(z_pos,start_contr_corr,folder,plot_equ=False,equ_z_pos=None
):
    plt.figure(figsize=(30,10))
    plt.plot(z_pos.T)
    for start in start_contr_corr:
        plt.plot([start,start],[0,np.nanmax(z_pos)+1],'k',linestyle=':')
        plt.plot([start,start],[0,np.nanmax(z_pos)+1],'k',linestyle=':')
    if plot_equ==True:
        for e in equ_z_pos:
            plt.plot([0,360],[e,e],'k',linestyle='--')
    plt.xlabel('time [frames]')
    plt.ylabel('z-line position [um]')
    plt.savefig(folder + '/z_pos_contr_equ.png',format= 'png')
def assign_peaks_old(z_pos,filename,time,start_idx=1,diff_min = 1.2,offset=0):
    folder =  os.path.splitext(filename)[0]
    fig_assign= plt.figure()
    ax_assign = fig_assign.add_subplot(1,1,1)
    z_pos_corr = z_pos.copy() *np.nan
    z_pos_corr[:,offset] = z_pos[:,offset]
    # delete all lines with nan's
    list_i  = range(start_idx,len(z_pos)) + list(np.arange(start_idx-1,-1,-1))
    for k,i in enumerate(list_i):
        z_pos_i = z_pos[i]
        if np.count_nonzero(np.isnan(z_pos_i)) == 0:
            for j,z in enumerate(z_pos_i):
                if j>0+offset:
                    try:
                        diff = np.abs(z_pos[:,j] - z_pos_corr[i,j-1])
                        argnext = np.nanargmin(diff)
                        #check if equal to upper z band
                        if z_pos[argnext,j]==z_pos_corr[i+1,j]:
                            argnext = argnext-1
                        #check if equal to lower z band
                        if z_pos[argnext,j]==z_pos_corr[i-1,j]:
                            argnext = argnext+1
                        #check if distance from last is bigger than 3
                        if True:
                            if np.abs(z_pos[argnext,j] - z_pos[list_i[k-1],j]) > 3:
```

```
                                if list_i[k]-list_i[k-1] > 0:
                                    argnext = argnext -1
                                elif list_i[k]-list_i[k-1] < 0:
                                    argnext = argnext +1
                            if np.abs(diff[argnext])>diff_min:
                                z_pos_corr[i,j] = z_pos_corr[i,j-1
                            else:
                                z_pos_corr[i,j] = z_pos[argnext,j]
                        except:
                            pass
            else:
                z_pos_corr[i] = z_pos_i * np.nan
        if len(time)== len(z_pos_corr.T):
            ax_assign.plot(time,z_pos_corr.T)
        if len(time)> len(z_pos_corr.T):
            ax_assign.plot(time[1:],z_pos_corr.T)
        if len(time)< len(z_pos_corr.T):
            ax_assign.plot(time,z_pos_corr.T[1:])
        ax_assign.set_ylabel('position in um')
        ax_assign.set_xlabel('time')
        fig_assign.savefig(folder+'/z_lines_position.png',format='png')
        plt.clf()
        plt.close('all')
        subprocess.Popen(['open',folder])
        #subprocess.Popen(['xdg-open',folder])
        return z_pos_corr, time
def sav_gol_filter(dat,windowlength=9,polyorder=4):
    filter_dat = sig-
nal.savgol_filter(dat,polyorder=polyorder,window_length=windowlength)
    return filter_dat
def order_parameter(phase,N_min=8):
    N = np.count_nonzero(~np.isnan(phase))
    if N>N_min:
        mean_field = 1/float(N) * np.nansum(np.exp(phase*1j))
    else:
        mean_field = np.nan
    return mean_field,np.abs(mean_field),np.angle(mean_field)
def most_freq_val(array,bins=30):
    range = (np.nanmin(array),np.nanmax(array))
    if np.any(np.isnan(range)) == False:
        a,b = np.histogram(array,bins=bins,range=range)
        val = b[np.argmax(a)]
        return val
    else:
        return np.nan
def weighted_avg_and_std(values, weights):
    idx = np.where(~np.isnan(values))
    values = values[idx]
    weights = weights[idx]
    """
    Return the weighted average and standard deviation.
    values, weights -- Numpy ndarrays with the same shape.
    """
    try:
        average = np.average(values, weights=weights)
        variance = np.average((values-average)**2, weights=weights)  # Fast and numeri-
cally precise
        return average, np.sqrt(variance)
    except:
        return np.nan,np.nan
def peakfinder(dat,thres,min_dist,thres_alt,thres_contr):
    idx_top = np.array(pu.indexes(-dat,thres,min_dist))
    idx_valley = np.array(pu.indexes(dat,thres=0.16))
    for i,idx in enumerate(idx_top):
        dist_valley = idx_valley- idx_top[i]
        dist_valley_forward = dist_valley.copy()
        dist_valley_forward[dist_valley_forward<0] = 100000
        arg_forward = np.nanargmin(dist_valley_forward)
        #dist_forward = np.nanmin(dist_valley_forward)
        altitude_forward = dat[idx_top[i]] - dat[idx_valley[arg_forward]]
        dist_valley_backwards = dist_valley.copy()
        dist_valley_backwards[dist_valley_backwards>0] = -100000
        arg_backwards = np.nanargmax(dist_valley_backwards)
        #dist_backwards = np.nanmax(dist_valley_backwards)
        altitude_backwards = dat[idx_top[i]] - dat[idx_valley[arg_backwards]]
        try:
            if altitude_backwards < -thres_alt and altitude_forward <- thres_alt:
                pass
```

```
            else:
                idx_top[i] = 0
        except:
            pass
    idx_pre = [x for x in idx_top if x<>0]
    #another threshold for contraction
    idx_ret = [x for x in idx_pre if dat[x]<0.2]
    return idx_ret
def manual_equ_array(delta_slen):
    equ_all = []
    for delta in delta_slen:
        equ = get_equillibrium_manual(delta)
        equ_all.append(equ)
    return equ_all
def get_equillibrium_manual(data):
    point_y = []
    xlim = (np.nanmin(data)*1.4,np.nanmax(data)*1.4)
    class LineBuilder:
        def __init__(self, line):
            self.line = line
            self.xs = list(line.get_xdata())
            self.ys = list(line.get_ydata())
            self.cid = line.figure.canvas.mpl_connect('button_press_event', self)
        def __call__(self, event):
            print('click', event)
            if event.inaxes!=self.line.axes: return
            #self.xs.append(event.xdata)
            #self.ys.append(event.ydata)
            #self.line.set_data(self.xs, self.ys)
            #self.line.figure.canvas.draw()
            point_y.append(event.ydata)
            ax.plot([-2,len(data)+2],[event.ydata,event.ydata],color='k',linestyle='--
',linewidth=2)
    fig = plt.figure(figsize=(16,5))
    ax = fig.add_subplot(111)
    ax.set_title('click to build equillibrium line')
    line, = ax.plot([0], [0])  # empty line
    linebuilder = LineBuilder(line)
    ax.set_ylim(xlim)
    ax.grid()
    ax.plot(data)
    plt.show()
    return point_y
def
get_heights_times_period(delta_slen_good,framerate,time,start_contr,scanrange_contr,fold
er,figsize=(30,5)):
    folder_contr = folder + '/contraction/'
    if os.path.isdir(folder_contr)== False:
        os.mkdir(folder + '/contraction/')
    start_contr_int = np.asarray([np.round(start,0) for start in start_contr])
    peaks_auto = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    peaks_auto_diff = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    height_auto = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    vel_minus = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    vel_plus = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    for j,delta in enumerate(delta_slen_good):
        plt.figure(figsize=figsize)
        plt.plot(time,delta,linewidth = 1.5,c='k')
        plt.xlabel('Time [s]')
        plt.ylabel('$\Delta$SL [um]')
        peaks_auto_i = start_contr_int*np.nan
        peaks_auto_diff_i = start_contr_int*np.nan
        height_auto_i = start_contr_int*np.nan
        vel_minus_i = start_contr_int*np.nan
        vel_plus_i = start_contr_int*np.nan
        for i,contr in enumerate(start_contr_int):
            plt.plot([time[contr],time[contr]],[np.nanmin(delta),np.nanmax(delta)],'--
',color='r',linewidth=1.5)
            delta_contr = del-
ta[contr:contr+np.round(float(framerate)/scanrange_contr,0)]
            vel = np.diff(delta_contr)
            try:
                #peaks =
np.nanmin(delta_contr)#peakfinder(delta_contr,thres=0.9,min_dist=0,thres_alt=0.0,thres_c
ontr=0.0)
                pos_peak_diff = np.nanargmin(delta_contr)
                pos_peak = np.nanargmin(delta_contr) + contr
            except:
```

```
                pos_peak_diff = np.nan
                pos_peak = np.nan
            #try:
            #    pos_peak = peaks[0] + contr
            #    pos_peak_diff = peaks[0]
            #except:
            #    pos_peak = peaks + contr
            #    pos_peak_diff = peaks
            #print pos_peak
                #print pos_peak
plt.plot([time[pos_peak],time[pos_peak]],[np.nanmin(delta),np.nanmax(delta)],'--
',color='k',linewidth=1.5)
            try:
                vel_minus_ji = np.nanmin(vel[:pos_peak_diff])
            except:
                vel_minus_ji = np.nan
            try:
                vel_plus_ji = np.nanmax(vel[pos_peak_diff:pos_peak_diff+12])
            except:
                try:
                    vel_plus_ji = np.nanmax(vel[pos_peak_diff:])
                except:
                    vel_plus_ji = np.nan
            peaks_auto_i[i] = pos_peak
            peaks_auto_diff_i[i] = pos_peak_diff
            height_auto_i[i] = delta[pos_peak]
            vel_minus_i[i] = vel_minus_ji
            vel_plus_i[i] = vel_plus_ji
        name_save = folder_contr + '/delta_slen_peaks_'+ str(j)
        plt.savefig(name_save + '.png',format='png')
        plt.savefig(name_save + '.eps',format='eps')
        peaks_auto[j] = peaks_auto_i
        peaks_auto_diff[j] = peaks_auto_diff_i
        height_auto[j] = height_auto_i
        vel_minus[j] = vel_minus_i
        vel_plus[j] = vel_plus_i
    np.save(folder +
'/time_height_auto.npy',[peaks_auto,peaks_auto_diff,height_auto,vel_minus,vel_plus])
    return peaks_auto,peaks_auto_diff,height_auto,vel_minus,vel_plus
def
get_heights_times_period_transl(delta_slen_good,framerate,time,start_contr,scanrange_con
tr,folder,figsize=(30,5)):
    folder_contr = folder + '/translation/'
    if os.path.isdir(folder_contr)== False:
        os.mkdir(folder + '/translation/')
    start_contr_int = np.asarray([np.round(start,0) for start in start_contr])
    peaks_auto = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    peaks_auto_diff = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    height_auto = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    vel_minus = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    vel_plus = np.zeros((len(delta_slen_good),len(start_contr_int)))*np.nan
    for j,delta in enumerate(delta_slen_good):
        plt.figure(figsize=figsize)
        plt.plot(time,delta,linewidth = 1.5,c='k')
        plt.xlabel('Time [s]')
        plt.ylabel('$\Delta$s [um]')
        peaks_auto_i = start_contr_int*np.nan
        peaks_auto_diff_i = start_contr_int*np.nan
        height_auto_i = start_contr_int*np.nan
        vel_minus_i = start_contr_int*np.nan
        vel_plus_i = start_contr_int*np.nan
        for i,contr in enumerate(start_contr_int):
            plt.plot([time[contr],time[contr]],[np.nanmin(delta),np.nanmax(delta)],'--
',color='r',linewidth=1.5)
            delta_contr = del-
ta[contr:contr+np.round(float(framerate)/scanrange_contr,0)]
            vel = np.diff(delta_contr)
            try:
                #peaks =
np.nanmin(delta_contr)#peakfinder(delta_contr,thres=0.9,min_dist=0,thres_alt=0.0,thres_c
ontr=0.0)
                pos_peak_diff = np.nanargmax(delta_contr)
                pos_peak = np.nanargmax(delta_contr) + contr
            except:
                pos_peak_diff = np.nan
                pos_peak = np.nan
            #try:
            #    pos_peak = peaks[0] + contr
```

```
            #    pos_peak_diff = peaks[0]
            #except:
            #    pos_peak = peaks + contr
            #    pos_peak_diff = peaks
            #print pos_peak
                #print pos_peak
plt.plot([time[pos_peak],time[pos_peak]],[np.nanmin(delta),np.nanmax(delta)],'--
',color='k',linewidth=1.5)
            try:
                vel_minus_ji = np.nanmin(vel[:pos_peak_diff])
            except:
                vel_minus_ji = np.nan
            try:
                vel_plus_ji = np.nanmax(vel[pos_peak_diff:pos_peak_diff+12])
            except:
                try:
                    vel_plus_ji = np.nanmax(vel[pos_peak_diff:])
                except:
                    vel_plus_ji = np.nan
            peaks_auto_i[i] = pos_peak
            peaks_auto_diff_i[i] = pos_peak_diff
            height_auto_i[i] = delta[pos_peak]-delta[pos_peak_diff]
            vel_minus_i[i] = vel_minus_ji
            vel_plus_i[i] = vel_plus_ji
        name_save = folder_contr + '/delta_transl_peaks_'+ str(j)
        plt.savefig(name_save + '.png',format='png')
        plt.savefig(name_save + '.eps',format='eps')
        peaks_auto[j] = peaks_auto_i
        peaks_auto_diff[j] = peaks_auto_diff_i
        height_auto[j] = height_auto_i
        vel_minus[j] = vel_minus_i
        vel_plus[j] = vel_plus_i
    np.save(folder +
'/time_height_auto_transl.npy',[peaks_auto,peaks_auto_diff,height_auto,vel_minus,vel_plu
s])
    return peaks_auto,peaks_auto_diff,height_auto,vel_minus,vel_plus
def get_x_manual(data,x_figsize):
    data = np.asarray(data)
    point_x = []
    ylim = (np.nanmin(np.asarray(data).T)*0.8,np.nanmax(np.asarray(data).T)*1.2)
    class LineBuilder:
        def __init__(self, line):
            self.line = line
            self.xs = list(line.get_xdata())
            self.ys = list(line.get_ydata())
            self.cid = line.figure.canvas.mpl_connect('button_press_event', self)
        def __call__(self, event):
            print('click', event)
            if event.inaxes!=self.line.axes: return
            #self.xs.append(event.xdata)
            #self.ys.append(event.ydata)
            #self.line.set_data(self.xs, self.ys)
            #self.line.figure.canvas.draw()
            point_x.append(event.xdata)
            ax.plot([event.xdata,event.xdata],[ylim[0],ylim[1]],color='k',linestyle='--
',linewidth=2)
    fig = plt.figure(figsize=(x_figsize,15))
    ax = fig.add_subplot(111)
    ax.set_title('click to build peaks')
    line, = ax.plot([0], [0])  # empty line
    linebuilder = LineBuilder(line)
    ax.set_ylim(ylim)
    ax.grid()
    ax.plot(data)
    plt.show()
    return point_x
def get_x_manual_small(data):
    data = np.asarray(data)
    point_x = []
    ylim = (np.nanmin(np.asarray(data).T)*0.8,np.nanmax(np.asarray(data).T)*1.2)
    class LineBuilder:
        def __init__(self, line):
            self.line = line
            self.xs = list(line.get_xdata())
            self.ys = list(line.get_ydata())
            self.cid = line.figure.canvas.mpl_connect('button_press_event', self)
        def __call__(self, event):
            print('click', event)
```

```
            if event.inaxes!=self.line.axes: return
            #self.xs.append(event.xdata)
            #self.ys.append(event.ydata)
            #self.line.set_data(self.xs, self.ys)
            #self.line.figure.canvas.draw()
            point_x.append(event.xdata)
            ax.plot([event.xdata,event.xdata],[ylim[0],ylim[1]],color='k',linestyle='--
',linewidth=2)
    fig = plt.figure(figsize=(13,6))
    ax = fig.add_subplot(111)
    ax.set_title('click to build peaks')
    line, = ax.plot([0], [0])  # empty line
    linebuilder = LineBuilder(line)
    ax.set_ylim(ylim)
    ax.grid()
    ax.plot(data)
    plt.show()
    return point_x
def
get_delta_slen(z_pos,folder,start_contr_corr,framerate,back_ind_rest=3,filtered=True,fil
terparams=(9,4),plot=False):
    if filtered==True:
        z_pos =
sav_gol_filter(z_pos,windowlength=filterparams[0],polyorder=filterparams[1])
    slen = np.asarray([np.diff(o) for o in np.asarray(z_pos).T]).T
    # determine sarcomere resting length
    equ = np.zeros(len(slen))*np.nan
    for i,sl in enumerate(slen):
        equ_i = []
        for j,start in enumerate(start_contr_corr):
            plt.plot([start,start],[np.nanmin(sl),np.nanmax(sl)])
            equ_i.append(np.nanmean(sl[start-back_ind_rest:start]))
        equ_i_mean = np.nanmean(equ_i)
        equ[i] = equ_i_mean
    delta_slen = np.asarray([slen[idx] - equ[idx] for idx in range(len(slen))])
    np.savetxt(folder+'/slen.txt',slen)
    np.savetxt(folder+'/delta_slen.txt',delta_slen)
    np.savetxt(folder+'/equ.txt',equ)
    back_ind_rest_param = [back_ind_rest]
    np.savetxt(folder + '/back_ind_rest_param.txt', back_ind_rest_param)
    # save period
    #period = most_freq_val(np.diff(start_contr_corr))/framerate
    #np.savetxt(folder + '/period.txt',period)
    try:
        period = most_freq_val(np.diff(start_contr_corr))/float(framerate)
        period_to_save=[period]
        np.savetxt(folder + '/period.txt',period_to_save)
    except:
        print 'ERROR: No period time'
    if plot==True:
        for i,delta in enumerate(delta_slen):
            plt.figure(figsize=(30,3))
            plt.grid()
            plt.plot(delta)
            for start in start_contr_corr:
                plt.plot([start,start],[np.nanmin(delta),np.nanmax(delta)])
            plt.plot([0,len(delta)],[0,0])
    return delta_slen,equ
def get_stereotypical_peaks(delta_slen, framerate, start_contr, folder, timeshift=0,
framerange=1):
    folder_stereotypical = folder + '/stereotypical'
    if os.path.exists(folder_stereotypical)==False:
        os.mkdir(folder_stereotypical)
    delta_slen_stereotyp = []
    for i,delta in enumerate(delta_slen):
        plt.figure()
        delta_all = []
        for start in start_contr:
            #if start+framerate < len(delta):
            if start+framerate*framerange < len(delta):
                plt.plot(delta[start+timeshift:start+timeshift+framerange*framerate])
                del-
ta_all.append(delta[start+timeshift:start+timeshift+framerange*framerate])

#plt.plot(delta[start+timeshift:start+timeshift+np.ceil(framerange*framerate)])
                #del-
ta_all.append(delta[start+timeshift:start+timeshift+np.ceil(framerange*framerate)])
```

```
        plt.plot([np.nanmean(np.asarray(delta_all).T[idx]) for idx in
range(int(framerange*framerate))],linewidth=5)
        #plt.plot([np.nanmean(np.asarray(delta_all).T[idx]) for idx in
range(int(np.ceil(framerange*framerate)))],linewidth=5)
        plt.grid()
        plt.xlabel('frames [idx]')
        plt.title('Sarcomere # '+str(i))
        plt.ylabel('$\Delta$SL')
        plt.savefig(folder_stereotypical + '/all_peaks_'+str(i)+'.png',format='png')
        print len(delta_all)
        delta_slen_stereotyp.append([np.nanmean(np.asarray(delta_all).T[idx]) for idx in
range(int(framerange*framerate))])
        #delta_slen_stereotyp.append([np.nanmean(np.asarray(delta_all).T[idx]) for idx
in range(int(np.ceil(framerange*framerate)))])
    delta_slen_stereotyp = np.asarray(delta_slen_stereotyp)
    for i,delta in enumerate(delta_slen_stereotyp):
        plt.figure()
        plt.plot(delta)
        plt.title('Sarcomere # '+str(i))
        plt.ylabel('$\Delta$SL')
        plt.xlabel('frames [idx]')

        plt.subplots_adjust(left=0.15)
        plt.grid()

        plt.savefig(folder_stereotypical +
'/delta_slen_ste_'+str(i)+'.png',format='png')
    plt.figure()
    time =
np.arange(0,(len(delta_slen_stereotyp.T))/float(framerate),1/float(framerate))
    print len(delta_slen_stereotyp.T)
    print len(time)
    plt.plot(time,delta_slen_stereotyp.T)
    plt.xlabel('time [s]')
    plt.ylabel('stereotypical sarcomer contraction [um]')
    plt.savefig(folder_stereotypical+'/stereotypical_peaks.png',format='png')
    plt.savefig(folder_stereotypical+'/stereotypical_peaks.eps',format='eps')
    np.savetxt(folder + '/delta_slen_stereotyp.txt',delta_slen_stereotyp)
    timeshift_framerange_param = [timeshift, framerange]
    np.savetxt(folder + '/timeshift_framerange_param.txt', timeshift_framerange_param)
    return delta_slen_stereotyp
def
make_correlation_plots(delta_slen_good,framerate,start_contr_int_correct,folder,t_lim=(0
,1),norm=False,timeshift=0):
    framerate_int =int(np.round(framerate,0))
    folder_corr = folder + '/correlation/'
    if os.path.isdir(folder_corr)== False:
        os.mkdir(folder + '/correlation/')
    #intervals = range(0,np.round(framerate_int*t_lim[1]))
    intervals = range(0,int(np.round(framerate_int*t_lim[1],0)))
    corr_dist_time =
np.zeros((len(intervals),len(start_contr_int_correct),len(delta_slen_good),len(delta_sle
n_good)))*np.nan
    for k,interval_k in enumerate(intervals):
        for i,start_i in enumerate(start_contr_int_correct):
            #choose right array of peak
            delta_k = np.asarray(delta_slen_good).T[start_i+timeshift+ inter-
val_k:start_i +timeshift+ interval_k+2].T
            for h,delta_h in enumerate(delta_k):
                for j,delta_j in enumerate(delta_k):
                    corr_hj = np.correlate(delta_h,delta_j)
                    corr_dist_time[k,i,h,j] = corr_hj
    corr_dist_time_final = np.zeros((len(intervals),len(corr_dist_time.T)))
    for i,interval_i in enumerate(intervals):
        corr_dist_time_i = corr_dist_time[interval_i]
        corr_dist_i_mean = np.zeros((len(corr_dist_time_i),len(delta_slen_good)))
        for j,spike_j in enumerate(corr_dist_time_i):
            corr_dist_j = np.zeros((len(delta_slen_good),len(delta_slen_good)*2))*np.nan
            for k in range(len(delta_slen_good)):
                norm = spike_j[k,k]
                for l in range(len(delta_slen_good)):
                    dist = abs(l-k)
                    val_written = False
                    corr_kl = spike_j[k,l]
                    if norm==True:
                        corr_kl = spike_j[k,l]/norm
                    for m in range(len(delta_slen_good)*2):
                        if np.isnan(corr_dist_j[dist,m])== True and val_written==False:
```

```
                            corr_dist_j[dist,m] = corr_kl
                            val_written = True
            corr_dist_j_mean = [np.nanmean(corr_dist_j[idx]) for idx in
range(len(corr_dist_j))]
            corr_dist_i_mean[j] = corr_dist_j_mean
        mean_dist = [np.nanmean(corr_dist_i_mean.T[idx]) for idx in
range(len(corr_dist_i_mean.T))]
        corr_dist_time_final[i] = mean_dist
    zp = []
    for interval in intervals:
            z = corr_dist_time_final[interval]
            #plt.plot(x,y,z)
            zp.append(z)
    #make plot 3d with rawdata and correlation
    plt.figure()
    zp = np.asarray(zp)
    xp,yp = np.meshgrid(range(len(delta_slen_good)+1),[float(r)/framerate_int for r in
range(len(zp))])
    #xp,yp = np.meshgrid(range(len(delta_slen_good)),[float(r)/framerate_int for r in
range(len(zp))])
    plt.subplot(1, 1, 1)
    vmax = np.nanmax(zp)
    plt.pcolor(xp, yp, zp, cmap='RdBu', vmin=-vmax, vmax=vmax)
    plt.xlim(0,7)
    plt.xlabel('Sarcomer distance [indices]')
    plt.ylabel('Time [s]')
    plt.colorbar(label='Correlation of contraction (normalized)')
    plt.savefig(folder_corr + '/corr_colorplot.png',format='png')
    plt.savefig(folder_corr + '/corr_colorplot.eps',format='eps'
    np.save(folder_corr + '/corr_dist_time.npy',corr_dist_time_final)
    plt.show()
        # velocity
    intervals = range(0,framerate_int)
    vel_good = [np.diff(delta_slen_good_i) for delta_slen_good_i in delta_slen_good]
    corr_dist_time =
np.zeros((len(intervals),len(start_contr_int_correct),len(vel_good),len(vel_good)))*np.n
an
    for k,interval_k in enumerate(intervals):
        for i,start_i in enumerate(start_contr_int_correct):
            #choose right array of peak
            delta_k = np.asarray(vel_good).T[start_i+ interval_k:start_i + inter-
val_k+2].T
            for h,delta_h in enumerate(delta_k):
                for j,delta_j in enumerate(delta_k):
                    corr_hj = np.correlate(delta_h,delta_j)
                    corr_dist_time[k,i,h,j] = corr_hj
    corr_dist_time_final = np.zeros((len(intervals),len(corr_dist_time.T)))
    for i,interval_i in enumerate(intervals):
        corr_dist_time_i = corr_dist_time[interval_i]
        corr_dist_i_mean = np.zeros((len(corr_dist_time_i),len(delta_slen_good)))
        for j,spike_j in enumerate(corr_dist_time_i):
            corr_dist_j = np.zeros((len(delta_slen_good),len(delta_slen_good)*2))*np.nan
            for k in range(len(delta_slen_good)):
                norm = spike_j[k,k]
                for l in range(len(delta_slen_good)):
                    dist = abs(l-k)
                    val_written = False
                    corr_kl = spike_j[k,l]
                    for m in range(len(delta_slen_good)*2):
                        if np.isnan(corr_dist_j[dist,m])== True and val_written==False:
                            corr_dist_j[dist,m] = corr_kl
                            val_written = True
            corr_dist_j_mean = [np.nanmean(corr_dist_j[idx]) for idx in
range(len(corr_dist_j))]
            corr_dist_i_mean[j] = corr_dist_j_mean
        mean_dist = [np.nanmean(corr_dist_i_mean.T[idx]) for idx in
range(len(corr_dist_i_mean.T))]
        corr_dist_time_final[i] = mean_dist
    zp = []
    for interval in intervals:
            z = corr_dist_time_final[interval]
            #plt.plot(x,y,z)
            zp.append(z)
    #make plot 3d with rawdata and correlation
    plt.figure()
    zp = np.asarray(zp)
    xp,yp = np.meshgrid(range(len(delta_slen_good)+1),[float(r)/framerate_int for r in
range(len(zp))])
```

```
    #xp,yp = np.meshgrid(range(len(delta_slen_good)),[float(r)/framerate_int for r in
range(len(zp))])
    plt.subplot(1, 1, 1)
    vmax = np.nanmax(zp)
    plt.pcolor(xp, yp, zp, cmap='RdBu', vmin=-vmax, vmax=vmax)
    plt.xlim(0,7)
    plt.xlabel('Sarcomer distance [indices]')
    plt.ylabel('Time [s]')
    plt.colorbar(label='Correlation of velocity (normalized)')

    plt.savefig(folder_corr + '/corr_colorplot_vel.png',format='png')
    plt.savefig(folder_corr + '/corr_colorplot_vel.eps',format='eps')
    np.save(folder_corr + '/corr_dist_time_vel.npy',corr_dist_time_final)
    plt.show()
def
make_correlation_plots_stereotyp(delta_slen_good,framerate,folder,t_lim=(0,1),norm=False
,timeshift=0):
    start_contr_int_correct = [0]

    framerate_int = int(np.round(framerate,0))

    folder_corr = folder + '/correlation/'

    if os.path.isdir(folder_corr)== False:
        os.mkdir(folder + '/correlation/')
    intervals = range(0,int(np.round(framerate_int*t_lim[1],0)))
    #intervals = range(0,np.round(framerate_int*t_lim[1]))
    #intervals = range(0,framerate_int+1)
    corr_dist_time =
np.zeros((len(intervals),len(start_contr_int_correct),len(delta_slen_good),len(delta_sle
n_good)))*np.nan
    for k,interval_k in enumerate(intervals):
        for i,start_i in enumerate(start_contr_int_correct):
            #choose right array of peak
            delta_k = np.asarray(delta_slen_good).T[start_i+timeshift+ inter-
val_k:start_i +timeshift+ interval_k+2].T
            for h,delta_h in enumerate(delta_k):
                for j,delta_j in enumerate(delta_k):
                    corr_hj = np.correlate(delta_h,delta_j)
                    corr_dist_time[k,i,h,j] = corr_hj
    corr_dist_time_final = np.zeros((len(intervals),len(corr_dist_time.T)))
    for i,interval_i in enumerate(intervals):
        corr_dist_time_i = corr_dist_time[interval_i]
        corr_dist_i_mean = np.zeros((len(corr_dist_time_i),len(delta_slen_good)))
        for j,spike_j in enumerate(corr_dist_time_i):
            corr_dist_j = np.zeros((len(delta_slen_good),len(delta_slen_good)*2))*np.nan
            for k in range(len(delta_slen_good)):
                norm = spike_j[k,k]
                for l in range(len(delta_slen_good)):
                    dist = abs(l-k)
                    val_written = False
                    corr_kl = spike_j[k,l]
                    if norm==True:
                        corr_kl = spike_j[k,l]/norm
                    for m in range(len(delta_slen_good)*2):
                        if np.isnan(corr_dist_j[dist,m])== True and val_written==False:
                            corr_dist_j[dist,m] = corr_kl
                            val_written = True
            corr_dist_j_mean = [np.nanmean(corr_dist_j[idx]) for idx in
range(len(corr_dist_j))]
            corr_dist_i_mean[j] = corr_dist_j_mean
        mean_dist = [np.nanmean(corr_dist_i_mean.T[idx]) for idx in
range(len(corr_dist_i_mean.T))]
        corr_dist_time_final[i] = mean_dist
    zp = []
    for interval in intervals:
            z = corr_dist_time_final[interval]
            #plt.plot(x,y,z)
            zp.append(z)
    #make plot 3d with rawdata and correlation
    plt.figure()
    zp = np.asarray(zp)
    xp,yp = np.meshgrid(range(len(delta_slen_good)+1),[float(r)/framerate_int for r in
range(len(zp))])
    #xp,yp = np.meshgrid(range(len(delta_slen_good)),[float(r)/framerate_int for r in
range(len(zp))])
    plt.subplot(1, 1, 1)
    vmax = np.max(zp.T[0:7])
```

```
    plt.pcolor(xp, yp, zp, cmap='RdBu', vmin=-vmax, vmax=vmax)
    plt.xlim(0,7)
    plt.ylim(t_lim[0],t_lim[1])
    plt.xlabel('Sarcomer distance [indices]')
    plt.ylabel('Time [s]')
    plt.title('Stereotypical peaks')
    plt.colorbar(label='Correlation of contraction (normalized)')
    plt.savefig(folder_corr + '/corr_colorplot_stereotyp.png',format='png')
    plt.savefig(folder_corr + '/corr_colorplot_stereotyp.eps',format='eps')
    np.save(folder_corr + '/corr_dist_time_stereotyp.npy',corr_dist_time_final)
    plt.show()
        # velocity
    intervals = range(0,int(np.round(framerate_int*t_lim[1],0)))
    #intervals = range(0,framerate_int+1)
    vel_good = [np.diff(delta_slen_good_i) for delta_slen_good_i in delta_slen_good]
    corr_dist_time =
np.zeros((len(intervals),len(start_contr_int_correct),len(vel_good),len(vel_good)))*np.n
an
    for k,interval_k in enumerate(intervals):
        for i,start_i in enumerate(start_contr_int_correct):
            #choose right array of peak
            delta_k = np.asarray(vel_good).T[start_i+ interval_k:start_i + inter-
val_k+2].T
            for h,delta_h in enumerate(delta_k):
                for j,delta_j in enumerate(delta_k):
                    corr_hj = np.correlate(delta_h,delta_j)
                    corr_dist_time[k,i,h,j] = corr_hj
    corr_dist_time_final = np.zeros((len(intervals),len(corr_dist_time.T)))
    for i,interval_i in enumerate(intervals):
        corr_dist_time_i = corr_dist_time[interval_i]
        corr_dist_i_mean = np.zeros((len(corr_dist_time_i),len(delta_slen_good)))
        for j,spike_j in enumerate(corr_dist_time_i):
            corr_dist_j = np.zeros((len(delta_slen_good),len(delta_slen_good)*2))*np.nan
            for k in range(len(delta_slen_good)):
                norm = spike_j[k,k]
                for l in range(len(delta_slen_good)):
                    dist = abs(l-k)
                    val_written = False
                    corr_kl = spike_j[k,l]
                    for m in range(len(delta_slen_good)*2):
                        if np.isnan(corr_dist_j[dist,m])== True and val_written==False:
                            corr_dist_j[dist,m] = corr_kl
                            val_written = True
            corr_dist_j_mean = [np.nanmean(corr_dist_j[idx]) for idx in
range(len(corr_dist_j))]
            corr_dist_i_mean[j] = corr_dist_j_mean
        mean_dist = [np.nanmean(corr_dist_i_mean.T[idx]) for idx in
range(len(corr_dist_i_mean.T))]
        corr_dist_time_final[i] = mean_dist
    zp = []
    for interval in intervals:
            z = corr_dist_time_final[interval]
            #plt.plot(x,y,z)
            zp.append(z)
    #make plot 3d with rawdata and correlation
    plt.figure()
    zp = np.asarray(zp)
    xp,yp = np.meshgrid(range(len(delta_slen_good)+1),[float(r)/framerate_int for r in
range(len(zp))])
    #xp,yp = np.meshgrid(range(len(delta_slen_good)),[float(r)/framerate_int for r in
range(len(zp))])
    plt.subplot(1, 1, 1)
    vmax = vmax = np.max(zp.T[0:7])
    plt.pcolor(xp, yp, zp, cmap='RdBu', vmin=-vmax, vmax=vmax)
    plt.xlim(0,7)
    plt.ylim(t_lim[0],t_lim[1])
    plt.xlabel('Sarcomer distance [indices]')
    plt.ylabel('Time [s]')
    plt.colorbar(label='Correlation of velocity (normalized)')
    plt.savefig(folder_corr + '/corr_colorplot_vel_stereotyp.png',format='png')
    plt.savefig(folder_corr + '/corr_colorplot_vel_stereotyp.eps',format='eps')
    np.save(folder_corr + '/corr_dist_time_vel_stereotyp.npy',corr_dist_time_final)
    plt.show()
def
make_correlation_plot_single_sarcomer(delta_slen_good,framerate,start_contr,folder,times
hift=0,norm=False):
    folder_corr = folder + '/correlation/'
    framerate_int = int(np.round(framerate,0))
```

```
    if os.path.isdir(folder_corr)== False:
        os.mkdir(folder + '/correlation/')
    for n in range(len(delta_slen_good)):
        intervals = range(0,framerate_int+1)
        corr_dist_time =
np.zeros((len(intervals),len(start_contr),len(delta_slen_good),len(delta_slen_good)))*np
.nan
        for k,interval_k in enumerate(intervals):
            for i,start_i in enumerate(start_contr):
                #choose right array of peak
                delta_k = np.asarray(delta_slen_good).T[start_i+timeshift+ inter-
val_k:start_i +timeshift+ interval_k+2].T
                for h,delta_h in enumerate(delta_k):
                    for j,delta_j in enumerate(delta_k):
                        corr_hj = np.correlate(delta_h,delta_j)
                        corr_dist_time[k,i,h,j] = corr_hj
        corr_dist_time_final = np.zeros((len(intervals),len(corr_dist_time.T)))*np.nan
        for i,interval_i in enumerate(intervals):
            corr_dist_time_i = corr_dist_time[interval_i]
            corr_dist_i_mean = np.zeros((len(corr_dist_time_i),len(delta_slen_good)))
            for j,spike_j in enumerate(corr_dist_time_i):
                corr_dist_j =
np.zeros((len(delta_slen_good),len(delta_slen_good)*2))*np.nan
                for l in range(len(delta_slen_good)):
                    dist = abs(l-n)
                    val_written = False
                    corr_nl = spike_j[n,l]
                    if norm==True:
                        corr_nl = spike_j[n,l]/norm
                    for m in range(len(delta_slen_good)*2):
                        if np.isnan(corr_dist_j[dist,m])== True and val_written==False:
                            corr_dist_j[dist,m] = corr_nl
                            val_written = True
                corr_dist_j_mean = [np.nanmean(corr_dist_j[idx]) for idx in
range(len(corr_dist_j))]
                corr_dist_i_mean[j] = corr_dist_j_mean
            mean_dist = [np.nanmean(corr_dist_i_mean.T[idx]) for idx in
range(len(corr_dist_i_mean.T))]
            corr_dist_time_final[i] = mean_dist
        zp = []
        for interval in intervals:
                z = corr_dist_time_final[interval]
                #plt.plot(x,y,z)
                zp.append(z)
        #make plot 3d with rawdata and correlation
        plt.figure()
        zp = np.asarray(zp)
        xp,yp = np.meshgrid(range(len(delta_slen_good)+1),[float(r)/framerate_int for r
in range(len(zp))])
        #xp,yp = np.meshgrid(range(len(delta_slen_good)),[float(r)/framerate_int for r
in range(len(zp))])
        plt.subplot(1, 1, 1)
        vmax = np.nanmax(zp)
        plt.pcolor(xp, yp, zp, cmap='RdBu', vmin=-vmax, vmax=vmax)
        plt.xlim(0,7)
        plt.xlabel('Sarcomer distance [indices]')
        plt.ylabel('Time [s]')
        plt.title('sarcomer #'+str(n))
        plt.colorbar(label='Correlation of contraction (normalized)')
        plt.savefig(folder_corr + '/corr_colorplot_sarc_'+str(n)+'.png',format='png')
        plt.savefig(folder_corr + '/corr_colorplot_sarc_'+str(n)+'.eps',format='eps')
        #np.save(folder + '/manual/corr_dist_time.npy',corr_dist_time_final)
        plt.show()
            # velocity
        intervals = range(0,framerate_int+1)
        vel_good = [np.diff(delta_slen_good_i) for delta_slen_good_i in delta_slen_good]
        corr_dist_time =
np.zeros((len(intervals),len(start_contr),len(vel_good),len(vel_good)))*np.nan
        for k,interval_k in enumerate(intervals):
            for i,start_i in enumerate(start_contr):
                #choose right array of peak
                delta_k = np.asarray(vel_good).T[start_i+ interval_k:start_i + inter-
val_k+2].T
                for h,delta_h in enumerate(delta_k):
                    for j,delta_j in enumerate(delta_k):
                        corr_hj = np.correlate(delta_h,delta_j)
                        corr_dist_time[k,i,h,j] = corr_hj
        corr_dist_time_final = np.zeros((len(intervals),len(corr_dist_time.T)))
```

```
        for i,interval_i in enumerate(intervals):
            corr_dist_time_i = corr_dist_time[interval_i]
            corr_dist_i_mean = np.zeros((len(corr_dist_time_i),len(delta_slen_good)))
            for j,spike_j in enumerate(corr_dist_time_i):
                corr_dist_j =
np.zeros((len(delta_slen_good),len(delta_slen_good)*2))*np.nan
                for l in range(len(delta_slen_good)):
                    dist = abs(l-n)
                    val_written = False
                    corr_nl = spike_j[n,l]
                    for m in range(len(delta_slen_good)*2):
                        if np.isnan(corr_dist_j[dist,m])== True and val_written==False:
                            corr_dist_j[dist,m] = corr_nl
                            val_written = True
                corr_dist_j_mean = [np.nanmean(corr_dist_j[idx]) for idx in
range(len(corr_dist_j))]
                corr_dist_i_mean[j] = corr_dist_j_mean
            mean_dist = [np.nanmean(corr_dist_i_mean.T[idx]) for idx in
range(len(corr_dist_i_mean.T))]
            corr_dist_time_final[i] = mean_dist
        zp = []
        for interval in intervals:
                z = corr_dist_time_final[interval]
                #plt.plot(x,y,z)
                zp.append(z)
        #make plot 3d with rawdata and correlation
        plt.figure()
        zp = np.asarray(zp)
        xp,yp = np.meshgrid(range(len(delta_slen_good)+1),[float(r)/framerate_int for r
in range(len(zp))])
        #xp,yp = np.meshgrid(range(len(delta_slen_good)),[float(r)/framerate_int for r
in range(len(zp))])
        plt.subplot(1, 1, 1)
        vmax = np.nanmax(zp)
        plt.pcolor(xp, yp, zp, cmap='RdBu', vmin=-vmax, vmax=vmax)
        plt.xlim(0,7)
        plt.xlabel('Sarcomer distance [indices]')
        plt.ylabel('Time [s]')
        plt.title('sarcomer #'+str(n))
        plt.colorbar(label='Correlation of velocity (normalized)')
        plt.savefig(folder_corr +
'/corr_colorplot_vel_sarc_'+str(n)+'.png',format='png')
        plt.savefig(folder_corr +
'/corr_colorplot_vel_sarc_'+str(n)+'.eps',format='eps')
        #np.save(folder + '/manual/corr_dist_time_vel.npy',corr_dist_time_final_vel)
        plt.show()
def
make_neighbor_correlation_plot(delta_slen_stereotyp,framerate,folder,idx_plot,yticks,xra
nge_neighbor=0.8,name_save=None,res_folder=None):
    delta_slen_ste = np.asarray([delta_slen_stereotyp[idx] for idx in idx_plot])
    #xrange_neighbor=int(xrange_neighbors)
    plt.figure(figsize=(3,10))
    ax = plt.subplot(1,1,1)
    ax.spines['top'].set_color('none')
    ax.spines['bottom'].set_color('none')
    ax.spines['left'].set_color('none')
    ax.spines['right'].set_color('none')
    ax.tick_params(labelcolor='w', top='off', bottom='off', left='off', right='off')
    time = np.arange(0,(len(delta_slen_ste.T))/float(framerate),1/float(framerate))
    len_delta = len(delta_slen_ste)
    ylim = [np.min(delta_slen_ste),np.max(delta_slen_ste)]
    tick = float(int(10*np.max(np.abs(ylim))))/10
    print tick
    #yticks = [-tick,0,tick]
    num_subplot = 8*3+7
    X = [
((1,3)),(4),((5,7)),(8),(9,11),12,(13,15),16,(17,19),20,(21,23),24,(25,27),28,(29,31)]
    for i,delta in enumerate(delta_slen_ste):
        ax1 = plt.subplot(num_subplot,1,X[2*i])
        ax1.grid()
        ax1.plot(time,delta_slen_ste[i],'k',linewidth=1.5)
        ax1.set_yticks(yticks)
        ax1.set_yticklabels(yticks, fontsize=12)
        ax1.set_xticklabels([])
        ax1.set_ylim(ylim)
        ax1.set_xlim(0,xrange_neighbor)
        #plt.ylim(-ylim,ylim)
        #plt.set_xlim(0,np.max(time))
```

```
        corr = delta.copy()*np.nan
        if i < len_delta-1:
            for j,d_t in enumerate(delta):
                d_1 = delta_slen_ste[i][j]
                d_2 = delta_slen_ste[i+1][j]
                corr[j] = d_1*d_2
            len_time = len(delta_slen_ste.T)
            xx = np.arange(-0.5,len_time-0.5)/framerate
            yy = range(2)
            XX,YY = np.meshgrid(xx,yy)
            ZZ = np.asarray([list(corr),list(corr)])
            v_max = np.nanmax(np.abs(ZZ))
            #plt.subplot(num_subplot,1,X[2*i+1])
            ax2 = plt.subplot(num_subplot,1,X[2*i+1])
            ax2.pcolor(XX,YY,ZZ,cmap='RdBu', vmin=-v_max/2, vmax=v_max/2)
            ax2.set_yticks([])
            ax2.set_yticks([])
            ax2.set_yticklabels([],fontsize=0)
            ax2.set_xticks([])
            ax2.set_xlim(0,xrange_neighbor)
            #ax2.set(aspect=1,)
    ax1.set_xlabel('time [s]')
    xticks_3 = np.arange(0,xrange_neighbor,0.2)
    #xticks_3 = [0,0.2,0.4,0.6,0.8]
    ax1.set_xticks(xticks_3)
    ax1.set_xticklabels(xticks_3,fontsize=12)
    plt.subplots_adjust(bottom=0.08, left=0.2, top = 0.95, right=0.95)
    #plt.show()
    ax.set_ylabel('stereotypic contraction [um]',color='k')
    plt.savefig(folder + '/correlation/neighborcorrelation.png',format='png')
    plt.savefig(folder + '/correlation/neighborcorrletaion.eps',format='eps')
    #if name_save<>None:
    #    plt.savefig(res_folder +'/'+ name_save + '.png',format='png')
    #    plt.savefig(res_folder +'/'+  name_save +'.eps',format='eps')
def
get_corrected_start_contr_by_correlation(z_pos_good,framerate,start_contr,shift_contr=0)
:
    shift_array = []
    start_contr_int = [np.round(start,0) for start in start_contr]
    len_interval = framerate
    start_contr_int_0 = [contr+shift_contr for contr in start_contr_int if
contr+len_interval < len(np.asarray(z_pos_good).T)]
    # determine shift correction of start time by crosscorrelation of peaks of each sar-
comer and average
    shift_contr = []
    #z_pos_equ_good  = [z - equ_z_pos[i] for i,z in enumerate(z_pos_good)]
    for z in z_pos_good:
        #fig,ax = plt.subplots(figsize=(10,4))
        z_cc = np.zeros((len(start_contr_int_0),len_interval))
        # reference idx k
        for i,contr in enumerate(start_contr_int_0):
            # cc is for crosscorrelation
            #ax.plot(z[contr:contr+len_interval])
            z_cc[i]=z[contr:contr+len_interval]
        for k,z_k in enumerate(z_cc):
            shift_contr_z = []
            for j,z_j in enumerate(z_cc):
                #print z_k,z_j
                shift =  len(z_k)-np.nanargmax(np.correlate(z_k,z_j,mode='full'))-1
                shift_contr_z.append(shift)
                #print shift
            shift_contr.append(shift_contr_z)
            shift_contr_mean_k = [np.round(np.median(np.asarray(shift_contr).T[idx]),0)
for idx in range(len(np.asarray(shift_contr).T))]
            #print shift_contr_mean_k
            shift_array.append(shift_contr_mean_k)
    shift_all = [np.round(np.nanmean(contr),0) for contr in np.asarray(shift_array).T]
    start_contr_int_correct = [start_contr_int_0[idx]+shift_all[idx] for idx in
range(len(start_contr_int_0))]
    return start_contr_int_correct
def convert_to_list(arr,framerate=1):
    arr_list = []
    for arr_i in arr:
        for arr_j in arr_i:
            if np.isnan(arr_j)==False:
                arr_list.append(arr_j/framerate)
    return arr_list
def make_hist_1cell(peaks_auto_diff,height_auto,vel_plus,vel_minus,framerate,folder):
```

```
    height_hist  = convert_to_list(height_auto)
    times_hist = convert_to_list(peaks_auto_diff,framerate=framerate)
    fig_hist = plt.figure(figsize=(12,9))
    ax1 = fig_hist.add_subplot(2,2,1)
    ax1.hist(height_hist)
    ax1.set_xlim(0,np.nanmin(height_hist))
    ax1.set_xlabel('$\Delta$SL$_{max}$ [um]')
    ax1.set_ylabel('# Contractions')
    ax1.set_title('Amplitude of Max. Contraction $\Delta$SL$_{max}$')
    ax2 = fig_hist.add_subplot(2,2,2)
    ax2.hist(times_hist)
    ax2.set_xlabel('t$_{max}$ [s]')
    ax2.set_ylabel('# Contractions')
    ax2.set_title('Time of Max. Contraction')
    ax3 = fig_hist.add_subplot(2,2,3)
    ax3.scatter(times_hist,height_hist)
    ax3.set_title('Amlplitude vs. Time of Max. Contr.')
    ax3.set_xlabel('t$_{max}$ [s]')
    ax3.set_ylabel('$\Delta$SL$_{max}$ [um]')
    ax3.grid()
    ax4 = fig_hist.add_subplot(2,2,4)
    x_ticks = ['+','-']
    factor = 1/float(framerate)
    data_boxplot =
[np.abs(convert_to_list(vel_minus,factor)),convert_to_list(vel_plus,factor)]
    ax4.boxplot(data_boxplot,labels = x_ticks)
    ax4.set_title('(+) an (-) Velocity')
    ax4.set_ylabel('Velocity [um/s]')
    plt.subplots_adjust(bottom=0.05,left=0.08,right=0.97,top=0.95)
    fig_hist.savefig(folder + '/histograms_1cell.png',format='png')
    fig_hist.savefig(folder + '/histograms_1cell.eps',format='eps')
def
make_hist_1cell_transl(peaks_auto_diff,height_auto,vel_plus,vel_minus,framerate,folder):
    height_hist  = convert_to_list(height_auto)
    times_hist = convert_to_list(peaks_auto_diff,framerate=framerate)
    fig_hist = plt.figure(figsize=(12,9))
    ax1 = fig_hist.add_subplot(2,2,1)
    ax1.hist(height_hist)
    ax1.set_xlim(0,np.nanmax(height_hist))
    ax1.set_xlabel('$\Delta$SL$_{max}$ [um]')
    ax1.set_ylabel('# Contractions')
    ax1.set_title('Amplitude of Max. Contraction $\Delta$SL$_{max}$')
    ax2 = fig_hist.add_subplot(2,2,2)
    ax2.hist(times_hist)
    ax2.set_xlabel('t$_{max}$ [s]')
    ax2.set_ylabel('# Contractions')
    ax2.set_title('Time of Max. Contraction')
    ax3 = fig_hist.add_subplot(2,2,3)
    ax3.scatter(times_hist,height_hist)
    ax3.set_title('Amlplitude vs. Time of Max. Contr.')
    ax3.set_xlabel('t$_{max}$ [s]')
    ax3.set_ylabel('$\Delta$SL$_{max}$ [um]')
    ax3.grid()
    ax4 = fig_hist.add_subplot(2,2,4)
    x_ticks = ['+','-']
    factor = 1/float(framerate)
    data_boxplot =
[np.abs(convert_to_list(vel_minus,factor)),convert_to_list(vel_plus,factor)]
    ax4.boxplot(data_boxplot,labels = x_ticks)
    ax4.set_title('(+) an (-) Velocity')
    ax4.set_ylabel('Velocity [um/s]')
    plt.subplots_adjust(bottom=0.05,left=0.08,right=0.97,top=0.95)
    fig_hist.savefig(folder + '/histograms_1cell.png',format='png')
    fig_hist.savefig(folder + '/histograms_1cell.eps',format='eps')
```

7 Bibliography

Agbulut O, Huet A, Niederländer N, Puceat M, Menasché P, Coirault C (2007): Green fluorescent protein impairs actin-myosin interactions by binding to the actin-binding site of myosin. J Biol Chem 282, 10465–10471

Alberts B, Wilson J, Hunt T, Johnson A (Hrsg.): Molecular biology of the cell. 5. Auflage; Garland Science, New York 2008

Ansari AM, Ahmed AK, Matsangos AE, Lay F, Born LJ, Marti G, Harmon JW, Sun Z (2016): Cellular GFP toxicity and immunogenicity: Potential confounders in in vivo cell tracking experiments. Stem Cell Rev 12, 553–559

Anson B (2015): Building richer assays. Genet Eng Biotechn N 35, 20–21

Atmanli A, Domian IJ (2017): Recreating the cardiac microenvironment in pluripotent stem cell models of human physiology and disease. Trends Cell Biol 27, 352–364

Baker BM, Chen CS (2012): Deconstructing the third dimension – how 3D culture microenvironments alter cellular cues. J Cell Sci 125, 3015–3024

Berry MF, Engler AJ, Woo YJ, Pirolli TJ, Bish LT, Jayasankar V, Morine KJ, Gardner TJ, Discher DE, Sweeney HL (2006): Mesenchymal stem cell injection after myocardial infarction improves myocardial compliance. Am J Physiol Heart Circ Physiol 290, H2196-2203

Bers DM (2002): Cardiac excitation-contraction coupling. Nature 415, 198–205

Bokník P, Neumann J, Kaspareit G, Schmitz W, Scholz H, Vahlensieck U, Zimmermann N (1997): Mechanisms of the contractile effects of levosimendan in the mammalian heart. J Pharmacol Exp Ther 280, 277–283

Boothe SD, Myers JD, Pok S, Sun J, Xi Y, Nieto RM, Cheng J, Jacot JG (2016): The effect of substrate stiffness on cardiomyocyte action potentials. Cell Biochem Biophys 74, 527–535

Braam SR, Tertoolen L, van de Stolpe A, Meyer T, Passier R, Mummery CL (2010): Prediction of drug-induced cardiotoxicity using human embryonic stem cell-derived cardiomyocytes. Stem Cell Res 4, 107–116

Bray MA, Sheehy SP, Parker KK (2008): Sarcomere alignment is regulated by myocyte shape. Cell Motil Cytoskeleton 65, 641–651

Burridge PW, Keller G, Gold JD, Wu JC (2012): Production of de novo cardiomyocytes: Human pluripotent stem cell differentiation and direct reprogramming. Cell Stem Cell 10, 16–28

Chen X, Zaro JL, Shen WC (2013): Fusion protein linkers: property, design and functionality. Adv Drug Deliv Rev 65, 1357–1369

Christalla P, Hudson JE, Zimmermann WH (2012): The cardiogenic niche as a fundamental building block of engineered myocardium. Cells Tissues Organs 195, 82–93

Chudakov DM, Matz MV, Lukyanov S, Lukyanov KA (2010): Fluorescent proteins and their applications in imaging living cells and tissues. Physiol Rev 90, 1103–1163

Cleland JGF, Teerlink JR, Senior R, Nifontov EM, Mc Murray JJV, Lang CC, Tsyrlin VA, Greenberg BH, Mayet J, Francis DP, et al. (2011): The effects of the cardiac myosin activator, omecamtiv mecarbil, on cardiac function in systolic heart failure: a double-blind, placebo-controlled, crossover, dose-ranging phase 2 trial. Lancet 378, 676–683

Cong L, Ran FA, Cox D, Lin S, Barretto R, Habib N, Hsu PD, Wu X, Jiang W, Marraffini LA, Zhang F (2013): Multiplex genome engineering using CRISPR/Cas systems. Science 339, 819–823

da Rocha AM, Campbell K, Mironov S, Jiang J, Mundada L, Guerrero-Serna G, Jalife J, Herron TJ (2017): hiPSC-CM Monolayer maturation state determines drug responsiveness in high throughput pro-arrhythmia screen. Sci Rep 7, 13834

Dabiri GA, Turnacioglu KK, Sanger JM, Sanger JW (1997): Myofibrillogenesis visualized in living embryonic cardiomyocytes. Proc Natl Acad Sci U S A 94, 9493–9498

Dasbiswas K, Majkut S, Discher DE, Safran SA (2015): Substrate stiffness-modulated registry phase correlations in cardiomyocytes map structural order to coherent beating. Nat Commun 6, 6085

del Álamo JC, Lemons D, Serrano R, Savchenko A, Cerignoli F, Bodmer R, Mercola M (2016): High throughput physiological screening of iPSC-derived cardiomyocytes for drug development. Biochim Biophys Acta 1863, 1717–1727

Doyle T, Botstein D (1996): Movement of yeast cortical actin cytoskeleton visualized in vivo. Proc Natl Acad Sci U S A 93, 3886–3891

Duval K, Grover H, Han L-H, Mou Y, Pegoraro AF, Fredberg J, Chen Z (2017): Modeling physiological events in 2D vs. 3D cell culture. Physiol Bethesda 32, 266–277

Engler AJ, Carag-Krieger C, Johnson CP, Raab M, Tang HY, Speicher DW, Sanger JW, Sanger JM, Discher DE (2008): Embryonic cardiomyocytes beat best on a matrix with heart-like elasticity: scar-like rigidity inhibits beating. J Cell Sci 121, 3794–3802

Fang Y, Eglen RM (2017): Three-dimensional cell cultures in drug discovery and development. SLAS Discov 22, 456–472

Fong AH, Romero-López M, Heylman CM, Keating M, Tran D, Sobrino A, Tran AQ, Pham HH, Fimbres C, Gershon PD, et al. (2016): Three-dimensional adult cardiac extracellular matrix promotes maturation of human induced pluripotent stem cell-derived cardiomyocytes. Tissue Eng Part A 22, 1016–1025

Frantz C, Stewart KM, Weaver VM (2010): The extracellular matrix at a glance. J Cell Sci 123, 4195–4200

Gabriel-Costa D (2018): The pathophysiology of myocardial infarction-induced heart failure. Pathophysiology 25, 277–284

Galie PA, Khalid N, Carnahan KE, Westfall MV, Stegemann JP (2013): Substrate stiffness affects sarcomere and costamere structure and electrophysiological function of isolated adult cardiomyocytes. Cardiovasc Pathol 22, 219–227

Gerdes AM (2002): Cardiac myocyte remodeling in hypertrophy and progression to failure. J Card Fail 8, S264-268

Gibson TJ, Seiler M, Veitia RA (2013): The transience of transient overexpression. Nat Methods 10, 715–721

Griesbeck O, Baird GS, Campbell RE, Zacharias DA, Tsien RY (2001): Reducing the environmental sensitivity of yellow fluorescent protein. Mechanism and applications. J Biol Chem 276, 29188–29194

Hampe N, Jonas T, Wolters B, Hersch N, Hoffmann B, Merkel R (2014): Defined 2-D microtissues on soft elastomeric silicone rubber using lift-off epoxy-membranes for biomechanical analyses. Soft Matter 10, 2431–2443

Hazeltine LB, Simmons CS, Salick MR, Lian X, Badur MG, Han W, Delgado SM, Wakatsuki T, Crone WC, Pruitt BL, Palecek SP (2012): Effects of substrate mechanics on contractility of cardiomyocytes generated from human pluripotent stem cells. Int J Cell Biol 2012, 508294

Hersch N, Wolters B, Dreissen G, Springer R, Kirchgeßner N, Merkel R, Hoffmann B (2013): The constant beat: cardiomyocytes adapt their forces by equal contraction upon environmental stiffening. Biol Open 2, 351–361

Houdusse A, Sweeney HL (2016): How myosin generates force on actin filaments. Trends Biochem Sci 41, 989–997

Hsu PD, Lander ES, Zhang F (2014): Development and applications of CRISPR-Cas9 for genome engineering. Cell 157, 1262–1278

Hudson J, Titmarsh D, Hidalgo A, Wolvetang E, Cooper-White J (2012): Primitive cardiac cells from human embryonic stem cells. Stem Cells Dev 21, 1513–1523

Itskovitz-Eldor J, Schuldiner M, Karsenti D, Eden A, Yanuka O, Amit M, Soreq H, Benvenisty N (2000): Differentiation of human embryonic stem cells into embryoid bodies compromising the three embryonic germ layers. Mol Med 6, 88–95

Kaneko S, Kakinuma S, Asahina Y, Kamiya A, Miyoshi M, Tsunoda T, Nitta S, Asano Y, Nagata H, Otani S, et al. (2016): Human induced pluripotent stem cell-derived hepatic cell lines as a new model for host interaction with hepatitis B virus. Sci Rep 6, 29358

Kehat I, Kenyagin-Karsenti D, Snir M, Segev H, Amit M, Gepstein A, Livne E, Binah O, Itskovitz-Eldor J, Gepstein L (2001): Human embryonic stem cells can differentiate into myocytes with structural and functional properties of cardiomyocytes. J Clin Invest 108, 407–414

Kelley LA, Mezulis S, Yates CM, Wass MN, Sternberg MJE (2015): The Phyre2 web portal for protein modeling, prediction and analysis. Nat Protoc 10, 845–858

Khan JM, Lyon AR, Harding SE (2013): The case for induced pluripotent stem cell-derived cardiomyocytes in pharmacological screening. Br J Pharmacol 169, 304–317

Kim C, Wong J, Wen J, Wang S, Wang C, Spiering S, Kan NG, Forcales S, Puri PL, Leone TC, et al. (2013): Studying arrhythmogenic right ventricular dysplasia with patient-specific iPSCs. Nature 494, 105–110

Kuo PL, Lee H, Bray MA, Geisse NA, Huang YT, Adams WJ, Sheehy SP, Parker KK (2012): Myocyte shape regulates lateral registry of sarcomeres and contractility. Am J Pathol 181, 2030–2037

Kurokawa YK, George SC (2016): Tissue engineering the cardiac microenvironment: Multicellular microphysiological systems for drug screening. Adv Drug Deliv Rev 96, 225–233

Liu H, Qin W, Shao Y, Ma Z, Ye T, Borg T, Gao BZ (2011): Myofibrillogenesis in live neonatal cardiomyocytes observed with hybrid two-photon excitation fluorescence-second harmonic generation microscopy. J Biomed Opt 16, 126012

Liu Y, White HD, Belknap B, Winkelmann DA, Forgacs E (2015): Omecamtiv Mecarbil modulates the kinetic and motile properties of porcine β-cardiac myosin. Biochemistry 54, 1963–1975

Louch WE, Sheehan KA, Wolska BM (2011): Methods in cardiomyocyte isolation, culture, and gene transfer. J Mol Cell Cardiol 51, 288–298

Lundy SD, Zhu WZ, Regnier M, Laflamme MA (2013): Structural and functional maturation of cardiomyocytes derived from human pluripotent stem cells. Stem Cells Dev 22, 1991–2002

Majkut S, Idema T, Swift J, Krieger C, Liu A, Discher DE (2013): Heart-specific stiffening in early embryos parallels matrix and myosin expression to optimize beating. Curr Biol 23, 2434–2439

Mali P, Yang L, Esvelt KM, Aach J, Guell M, DiCarlo JE, Norville JE, Church GM (2013): RNA-guided human genome engineering via Cas9. Science 339, 823–826

Malik FI, Hartman JJ, Elias KA, Morgan BP, Rodriguez H, Brejc K, Anderson RL, Sueoka SH, Lee KH, Finer JT, et al. (2011): Cardiac myosin activation: a potential therapeutic approach for systolic heart failure. Science 331, 1439–1443

McCain ML, Yuan H, Pasqualini FS, Campbell PH, Parker KK (2014): Matrix elasticity regulates the optimal cardiac myocyte shape for contractility. Am J Physiol Heart Circ Physiol 306, H1525-1539

McDevitt TC, Angello JC, Whitney ML, Reinecke H, Hauschka SD, Murry CE, Stayton PS (2002): In vitro generation of differentiated cardiac myofibers on micropatterned laminin surfaces. J Biomed Mater Res 60, 472–479

Menasché P, Vanneaux V, Hagège A, Bel A, Cholley B, Cacciapuoti I, Parouchev A, Benhamouda N, Tachdjian G, Tosca L, et al. (2015): Human embryonic stem cell-derived cardiac progenitors for severe heart failure treatment: first clinical case report. Eur Heart J 36, 2011–2017

Moretti A, Bellin M, Welling A, Jung CB, Lam JT, Bott-Flügel L, Dorn T, Goedel A, Höhnke C, Hofmann F, et al. (2010): Patient-specific induced pluripotent stem-cell models for long-QT syndrome. N Engl J Med 363, 1397–1409

Morgan BP, Muci A, Lu PP, Qian X, Tochimoto T, Smith WW, Garard M, Kraynack E, Collibee S, Suehiro I, et al. (2010): Discovery of omecamtiv mecarbil the first, selective, small molecule activator of cardiac Myosin. ACS Med Chem Lett 1, 472–477

Nagy L, Kovács Á, Bódi B, Pásztor ET, Fülöp GÁ, Tóth A, Édes I, Papp Z (2015): The novel cardiac myosin activator omecamtiv mecarbil increases the calcium sensitivity of force production in isolated cardiomyocytes and skeletal muscle fibres of the rat. Br J Pharmacol 172, 4506–4518

Navarrete EG, Liang P, Lan F, Sanchez-Freire V, Simmons C, Gong T, Sharma A, Burridge PW, Patlolla B, Lee AS, et al. (2013): Screening adverse drug-induced arrhythmia events using human induced pluripotent stem cell-derived cardiomyocytes and low-impedance microelectrode arrays. Circulation 128

Nishimura S, Nagai S, Sata M, Katoh M, Yamashita H, Saeki Y, Nagai R, Sugiura S (2006): Expression of green fluorescent protein impairs the force-generating ability of isolated rat ventricular cardiomyocytes. Mol Cell Biochem 286, 59–65

Oikonomopoulos A, Kitani T, Wu JC (2018): Pluripotent stem cell-derived cardiomyocytes as a platform for cell therapy applications: Progress and hurdles for clinical translation. Mol Ther J Am Soc Gene Ther 26, 1624–1634

Parker KK, Tan J, Chen CS, Tung L (2008): Myofibrillar architecture in engineered cardiac myocytes. Circ Res 103, 340–342

Pasqualin C, Gannier F, Yu A, Malécot CO, Bredeloux P, Maupoil V (2016): SarcOptiM for ImageJ: High-frequency online sarcomere length computing on stimulated cardiomyocytes. Am J Physiol Cell Physiol 311, C277-283

Perán M, Marchal JA, López E, Jiménez-Navarro M, Boulaiz H, Rodríguez-Serrano F, Carrillo E, Sánchez-Espin G, de Teresa E, Tosh D, Aranega A (2010): Human cardiac tissue induces transdifferentiation of adult stem cells towards cardiomyocytes. Cytotherapy 12, 332–337

Ponikowski P, Voors AA, Anker SD, Bueno H, Cleland JGF, Coats AJS, Falk V, González-Juanatey JR, Harjola VP, Jankowska EA, et al. (2016): 2016 ESC Guidelines for the diagnosis and treatment of acute and chronic heart failure. Eur Heart J 37, 2129–2200

Prendergast FG, Mann KG (1978): Chemical and physical properties of aequorin and the green fluorescent protein isolated from Aequorea forskålea. Biochemistry 17, 3448–3453

Pyle WG, Solaro RJ (2004): At the crossroads of myocardial signaling: The role of Z-discs in intracellular signaling and cardiac function. Circ Res 94, 296–305

Qin D, Xia Y, Whitesides GM (2010): Soft lithography for micro- and nanoscale patterning. Nat Protoc 5, 491–502

Ratz M, Testa I, Hell SW, Jakobs S (2015): CRISPR/Cas9-mediated endogenous protein tagging for RESOLFT super-resolution microscopy of living human cells. Sci Rep 5, 9592

Ribeiro AJS, Ang Y-S, Fu JD, Rivas RN, Mohamed TMA, Higgs GC, Srivastava D, Pruitt BL (2015): Contractility of single cardiomyocytes differentiated from pluripotent stem cells depends on physiological shape and substrate stiffness. Proc Natl Acad Sci U S A 112, 12705–12710

Ribeiro E de A, Pinotsis N, Ghisleni A, Salmazo A, Konarev PV, Kostan J, Sjöblom B, Schreiner C, Polyansky AA, Gkougkoulia EA, et al. (2014): The structure and regulation of human muscle α-actinin. Cell 159, 1447–1460

Ribeiro MC, Tertoolen LG, Guadix JA, Bellin M, Kosmidis G, D'Aniello C, Monshouwer-Kloots J, Goumans MJ, Wang YL, Feinberg AW, et al. (2015): Functional maturation of human pluripotent stem cell derived cardiomyocytes in vitro--correlation between contraction force and electrophysiology. Biomaterials 51, 138–150

Ribeiro MC, Slaats RH, Schwach V, Rivera-Arbelaez JM, Tertoolen LGJ, van Meer BJ, Molenaar R, Mummery CL, Claessens MMAE, Passier R (2020): A cardiomyocyte show of force: A fluorescent alpha-actinin reporter line sheds light on human cardiomyocyte contractility versus substrate stiffness. J Mol Cell Cardiol 141, 54–64

Riegler J, Tiburcy M, Ebert A, Tzatzalos E, Raaz U, Abilez OJ, Shen Q, Kooreman NG, Neofytou E, Chen VC, et al. (2015): Human engineered heart muscles engraft and survive long term in a rodent myocardial infarction model. Circ Res 117, 720–730

Roberts B, Haupt A, Tucker A, Grancharova T, Arakaki J, Fuqua MA, Nelson A, Hookway C, Ludmann SA, Mueller IA, et al. (2017): Systematic gene tagging using CRISPR/Cas9 in human stem cells to illuminate cell organization. Mol Biol Cell 28, 2854–2874

Robertson C, Tran DD, George SC (2013): Concise review: Maturation phases of human pluripotent stem cell-derived cardiomyocytes. Stem Cells 31, 829–837

Ronaldson-Bouchard K, Ma SP, Yeager K, Chen T, Song L, Sirabella D, Morikawa K, Teles D, Yazawa M, Vunjak-Novakovic G (2018): Advanced maturation of human cardiac tissue grown from pluripotent stem cells. Nature 556, 239–243

Russell B, Curtis MW, Koshman YE, Samarel AM (2010): Mechanical stress-induced sarcomere assembly for cardiac muscle growth in length and width. J Mol Cell Cardiol 48, 817–823

Salick MR, Napiwocki BN, Sha J, Knight GT, Chindhy SA, Kamp TJ, Ashton RS, Crone WC (2014): Micropattern width dependent sarcomere development in human ESC-derived cardiomyocytes. Biomaterials 35, 4454–4464

Sander JD, Joung JK (2014): CRISPR-Cas systems for genome editing, regulation and targeting. Nat Biotechnol 32, 347–355

Schaaf S, Shibamiya A, Mewe M, Eder A, Stöhr A, Hirt MN, Rau T, Zimmermann WH, Conradi L, Eschenhagen T, Hansen A (2011): Human engineered heart tissue as a versatile tool in basic research and preclinical toxicology. PloS One 6, e26397

Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, Preibisch S, Rueden C, Saalfeld S, Schmid B, et al. (2012): Fiji: An open-source platform for biological-image analysis. Nat Methods 9, 676–682

Schlosser F: Mechanics of suspended cells probed by dual optical traps in a confocal microscope. Phys. Diss. Göttingen 2015

Severs NJ (2000): The cardiac muscle cell. Bioessays 22, 188–199

Sharma A, Toepfer CN, Schmid M, Garfinkel AC, Seidman CE (2018): Differentiation and contractile analysis of GFP-sarcomere reporter hiPSC-cardiomyocytes. Curr Protoc Hum Genet 96, 21.12.1-21.12.12

Shen X, Soibam B, Benham A, Xu X, Chopra M, Peng X, Yu W, Bao W, Liang R, Azares A, et al. (2016): miR-322/-503 cluster is expressed in the earliest cardiac progenitor cells and drives cardiomyocyte specification. Proc Natl Acad Sci U S A 113, 9551–9556

Shimomura O, Johnson FH, Saiga Y (1962): Extraction, purification and properties of aequorin, a bioluminescent protein from the luminous hydromedusan, Aequorea. J Cell Comp Physiol 59, 223–239

Shintani SA, Oyama K, Kobirumaki-Shimozawa F, Ohki T, Ishiwata S, Fukuda N (2014): Sarcomere length nanometry in rat neonatal cardiomyocytes expressed with α-actinin-AcGFP in Z discs. J Gen Physiol 143, 513–524

Swenson AM, Tang W, Blair CA, Fetrow CM, Unrath WC, Previs MJ, Campbell KS, Yengo CM (2017): Omecamtiv Mecarbil enhances the duty ratio of human β-cardiac myosin resulting in increased calcium sensitivity and slowed force development in cardiac muscle. J Biol Chem 292, 3768–3778

Tachibana A, Santoso MR, Mahmoudi M, Shukla P, Wang L, Bennett M, Goldstone AB, Wang M, Fukushi M, Ebert AD, et al. (2017): Paracrine effects of the pluripotent stem cell-derived cardiac myocytes salvage the injured myocardium. Circ Res 121, e22–e36

Takahashi K, Tanabe K, Ohnuki M, Narita M, Ichisaka T, Tomoda K, Yamanaka S (2007): Induction of pluripotent stem cells from adult human fibroblasts by defined factors. Cell 131, 861–872

Teerlink JR, Clarke CP, Saikali KG, Lee JH, Chen MM, Escandon RD, Elliott L, Bee R, Habibzadeh MR, Goldman JH, et al. (2011): Dose-dependent augmentation of cardiac systolic function with the selective cardiac myosin activator, omecamtiv mecarbil: a first-in-man study. Lancet 378, 667–675

Teerlink JR, Felker GM, McMurray JJV, Solomon SD, Adams KF, Cleland JGF, Ezekowitz JA, Goudev A, Macdonald P, Metra M, et al. (2016): Chronic Oral Study of Myosin Activation to Increase Contractility in Heart Failure (COSMIC-HF): a phase 2, pharmacokinetic, randomised, placebo-controlled trial. Lancet 388, 2895–2903

Teerlink JR, Diaz R, Felker GM, McMurray JJV, Metra M, Solomon SD, Legg JC, Büchele G, Varin C, Kurtz CE, et al. (2020): Omecamtiv Mecarbil in chronic heart failure with reduced ejection fraction: Rationale and design of GALACTIC-HF. JACC Heart Fail 8, 329–340

Thery M, Piel M (2009): Adhesive micropatterns for cells: A microcontact printing protocol. Cold Spring Harb Protoc 4, 5255

Thomson JA, Itskovitz-Eldor J, Shapiro SS, Waknitz MA, Swiergiel JJ, Marshall VS, Jones JM (1998): Embryonic stem cell lines derived from human blastocysts. Science 282, 1145–1147

Tiburcy M, Zimmermann WH (2014): Modeling myocardial growth and hypertrophy in engineered heart muscle. Trends Cardiovasc Med 24, 7–13

Tiburcy M, Meyer T, Soong PL, Zimmermann WH (2014): Collagen-based engineered heart muscle. Methods Mol Biol 1181, 167–176

Tiburcy M, Hudson JE, Balfanz P, Schlick S, Meyer T, Chang Liao M-L, Levent E, Raad F, Zeidler S, Wingender E, et al. (2017): Defined engineered human myocardium with advanced maturation for applications in heart failure modeling and repair. Circulation 135, 1832–1847

Toller WG, Stranz C (2006): Levosimendan, a new inotropic and vasodilator agent. Anesthesiology 104, 556–569

Tulloch NL, Muskheli V, Razumova MV, Korte FS, Regnier M, Hauch KD, Pabon L, Reinecke H, Murry CE (2011): Growth of engineered human myocardium with mechanical loading and vascular co-culture. Circ Res 109, 47–59

Uemura T, Mori T, Kurihara T, Kawase S, Koike R, Satoga M, Cao X, Li X, Yanagawa T, Sakurai T, et al. (2016): Fluorescent protein tagging of endogenous protein in brain neurons using CRISPR/Cas9-mediated knock-in and in utero electroporation techniques. Sci Rep 6, 35861

Uosaki H, Cahan P, Lee DI, Wang S, Miyamoto M, Fernandez L, Kass DA, Kwon C (2015): Transcriptional landscape of cardiomyocyte maturation. Cell Rep 13, 1705–1716

van Mil A, Balk GM, Neef K, Buikema JW, Asselbergs FW, Wu SM, Doevendans PA, Sluijter JPG (2018): Modelling inherited cardiac disease using human induced pluripotent stem cell-derived cardiomyocytes: Progress, pitfalls, and potential. Cardiovasc Res 114, 1828–1842

Vu T, Ma P, Xiao JJ, Wang YMC, Malik FI, Chow AT (2015): Population pharmacokinetic-pharmacodynamic modeling of omecamtiv mecarbil, a cardiac myosin activator, in healthy volunteers and patients with stable heart failure. J Clin Pharmacol 55, 1236–1247

Wang H, La Russa M, Qi LS (2016): CRISPR/Cas9 in genome editing and beyond. Annu Rev Biochem 85, 227–264

Wang X, Tucker NR, Rizki G, Mills R, Krijger PH, de Wit E, Subramanian V, Bartell E, Nguyen XX, Ye J, et al. (2016): Discovery and validation of sub-threshold genome-wide association study loci using epigenomic signatures. Elife 5, e10557

Woody MS, Greenberg MJ, Barua B, Winkelmann DA, Goldman YE, Ostap EM (2018): Positive cardiac inotrope omecamtiv mecarbil activates muscle despite suppressing the myosin working stroke. Nat Commun 9, 3838

Xia Y, Whitesides GM (1998): Soft lithography. Angew Chem Int Ed Engl 37, 550–575

Yang H, Schmidt LP, Wang Z, Yang X, Shao Y, Borg TK, Markwald R, Runyan R, Gao BZ (2016): Dynamic myofibrillar remodeling in live cardiomyocytes under static stretch. Sci Rep 6, 20674

Yang X, Pabon L, Murry CE (2014): Engineering adolescence: Maturation of human pluripotent stem cell-derived cardiomyocytes. Circ Res 114, 511–523

Yu J, Vodyanik MA, Smuga-Otto K, Antosiewicz-Bourget J, Frane JL, Tian S, Nie J, Jonsdottir GA, Ruotti V, Stewart R, et al. (2007): Induced pluripotent stem cell lines derived from human somatic cells. Science 318, 1917–1920

Yuan H, Marzban B, Kit Parker K (2017): Myofibrils in cardiomyocytes tend to assemble along the maximal principle stress directions. J Biomech Eng 139, 121010

Zemel A, Rehfeldt F, Brown AEX, Discher DE, Safran SA (2010): Optimal matrix rigidity for stress fiber polarization in stem cells. Nat Phys 6, 468–473

Zhang D, Shadrin IY, Lam J, Xian HQ, Snodgrass HR, Bursac N (2013): Tissue-engineered cardiac patch for advanced functional maturation of human ESC-derived cardiomyocytes. Biomaterials 34, 5813–5820

Zhang J, Wilson GF, Soerens AG, Koonce CH, Yu J, Palecek SP, Thomson JA, Kamp TJ (2009): Functional cardiomyocytes derived from human induced pluripotent stem cells. Circ Res 104, e30-41

Zuppinger C (2016): 3D culture for cardiac cells. Biochim Biophys Acta 1863, 1873–1881

Zwi L, Caspi O, Arbel G, Huber I, Gepstein A, Park IH, Gepstein L (2009): Cardiomyocyte differentiation of human induced pluripotent stem cells. Circulation 120, 1513–1523

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich bezüglich dieser Dissertation unterstützt und gefördert haben.

Bedanken möchte ich mich insbesondere bei meinem Doktorvater Herrn Prof. Wolfram-Hubertus Zimmermann sowie bei Herrn Prof. Christoph Schmidt, welche dieses Projekt ermöglicht haben und mich dabei stets mit inspirierenden Ideen und Ratschlägen unterstützt haben.

Mein gebührender Dank geht an Herrn Dr. Malte Tiburcy für die allzeit überaus gute Betreuung und all seinen wissenschaftlichen Input.

Ein großer Dank geht an Herrn Daniel Haertter, ohne den die automatisierte Datenauswertung nicht möglich gewesen.

Herzlich bedanken möchte ich mich bei Dr. Eugenia Butkevich für ihre Hilfe bei jeglichen biochemikalischen Fragestellungen. Ebenfalls herzlich bedanken möchte ich mich bei Dr. Susanne Schlick für all ihre Unterstützung am Institut für Pharmakologie und Toxikologie.

Vielen Dank auch an Krasimira Sharkova, Daria Reher und Iris Quentin für ihre Hilfe bei der Zellkultur und der Differenzierung der Stammzellen zu Kardiomyozyten. Vielen Dank zudem an Tanja Gall, Ulrike Schulz und Kerstin von Roden für ihre Hilfe bei diversen biochemikalischen Untersuchungen.

Vielen Dank!